Theoretical Computer Science 412 (2011) 6135–6192

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Grammar semantics, analysis and parsing by abstract interpretation

Patrick Cousot [a,b,c,d,*], Radhia Cousot [b,c,d]

[a] Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012, United States
[b] École normale supérieure, 45 rue d'Ulm, 75230 Paris Cedex 05, France
[c] CNRS, Paris, France
[d] INRIA, Paris-Rocquencourt, France[1]

ARTICLE INFO

Article history:
Received 2 March 2007
Received in revised form 23 May 2011
Accepted 4 June 2011
Communicated by D. Sannella

Keywords:
Abstract interpretation
Context-free grammar
Bottom-up semantics
Top-down semantics
Abstract semantics
Grammar flow analysis
Grammar problem
Parsing

ABSTRACT

We study abstract interpretations of a fixpoint protoderivation semantics defining the maximal derivations of a transitional semantics of context-free grammars akin to pushdown automata. The result is a hierarchy of bottom-up or top-down semantics refining the classical equational and derivational language semantics and including Knuth grammar problems, classical grammar flow analysis algorithms and parsing algorithms.

1. Introduction

Grammar flow problems consist in computing a function of the [proto]language generated by the grammar for each nonterminal. This includes Knuth's grammar problem [1,2], grammar decision problems such as emptiness and finiteness [3], and classical compilation algorithms such as First and Follow [4]. For the later case, Ulrich Möncke and Reinhard Wilhelm introduced *grammar flow analysis* to solve computation problems over context-free grammars [5–7], [8, Section 8.2.4]. The idea is to provide two fixpoint algorithm schemata, one for bottom-up grammar flow analysis and one for top-down grammar flow analysis which can be instantiated with different parameters to get classical iterative algorithms such as First and Follow.

More generally, we show that grammar flow algorithms are abstract interpretations [9] of a hierarchy of bottom-up or top-down grammar semantics refining the classical (proto-)language semantics.

Then, we apply this comprehensive abstract-interpretation-based approach to the systematic derivation of parsing algorithms.

The mathematical background and the necessary elements of abstract interpretation are reminded in Appendix.

2. Languages

Let $\mathcal{A}$ be an *alphabet*, that is a finite set of *letters*. A *sentence* $\sigma \in \mathcal{A}^\star$ *over the alphabet* $\mathcal{A}$ of length $|\sigma| \triangleq n \geqslant 0$ is a possibly empty finite sequence $\sigma_1\sigma_2 \ldots \sigma_n$ of letters $\sigma_1, \sigma_2, \ldots, \sigma_n \in \mathcal{A}$. For $n = 0$, the empty sentence is denoted ϵ of length $|\epsilon| = 0$.

* Corresponding author at: Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012, United States.
1 Project-team "Abstraction" of INRIA common to CNRS and École normale supérieure.

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2011.06.005

A *language* Σ *over the alphabet* $\mathcal{A}$ is a set of sentences $\Sigma \in \wp(\mathcal{A}^\star)$. We represent concatenation by juxtaposition. It is extended to languages as $\Sigma\Sigma' \triangleq \{\sigma\sigma' \mid \sigma \in \Sigma \wedge \sigma' \in \Sigma'\}$. For brevity, σ denotes the language $\{\sigma\}$ so that we can write $\Sigma\sigma\Sigma'$ for $\Sigma\{\sigma\}\Sigma'$. The *junction* of languages is $\Sigma \fatsemi \Sigma' \triangleq \{\sigma_1\sigma_2\ldots\sigma_m\sigma'_2\ldots\sigma'_n \mid \sigma_1\sigma_2\ldots\sigma_m \in \Sigma \wedge \sigma'_1\sigma'_2\ldots\sigma'_n \in \Sigma' \wedge \sigma_m = \sigma'_1\}$. Given a set $\mathscr{P} \triangleq \{[_i \mid i \in \Delta\} \cup \{]_i \mid i \in \Delta\}$ of matching parentheses and an alphabet $\mathcal{A}$, the *Dyck language* $\mathbb{D}_{\mathscr{P},\mathcal{A}} \subseteq (\mathscr{P} \cup \mathcal{A})^\star$ over $\mathscr{P}$ and $\mathcal{A}$ is the set of well-parenthesized sentences over $\mathscr{P} \cup \mathcal{A}$. In any sentence $\sigma \in \mathbb{D}_{\mathscr{P},\mathcal{A}}$ the number of opening parentheses $[_i$ for $i \in \Delta$ is equal to the number of matching closing parentheses $]_i$ while in any prefix of σ there are more opening parentheses than closing parentheses. It is *pure* if $\mathcal{A} = \emptyset$. The *parenthesized language* over $\mathscr{P}$ and $\mathcal{A}$ is $\mathbb{P}_{\mathscr{P},\mathcal{A}} \triangleq \{[_i\sigma]_i \mid i \in \Delta \wedge \sigma \in \mathbb{D}_{\mathscr{P},\mathcal{A}} \setminus \{\epsilon\}\}$.

3. Context-free grammars

A context-free grammar [10,11] is a quadruple $\mathcal{G} = \langle \mathscr{T},\ \mathscr{N},\ \overline{S},\ \mathscr{R}\rangle$ where $\mathscr{T}$ is the alphabet of *terminals*, $\mathscr{N}$ such that $\mathscr{T} \cap \mathscr{N} = \emptyset$ is the alphabet of *nonterminals*, $\overline{S} \in \mathscr{N}$ is the *start symbol* (or *axiom*) and $\mathscr{R} \in \wp(\mathscr{N} \times \mathscr{V}^\star)$ is the finite set of *rules* written $A \to \sigma$ where the *left-hand side* $A \in \mathscr{N}$ is a nonterminal and the *right-hand side* $\sigma \in \mathscr{V}^\star$ is a possibly empty sentence over the *vocabulary* $\mathscr{V} \triangleq \mathscr{T} \cup \mathscr{N}$. By convention, the empty sentence ϵ does not belong to the vocabulary, $\epsilon \notin \mathscr{V}$.

Example 1. $\langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$ is a grammar. $\square$

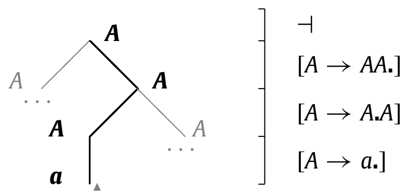
4. Transitional semantics of context-free grammars

Pushdown automata (PDA) are a classical language recognition mechanism first introduced by Oettinger in 1961 [12].[2] They are essentially finite state automata that can use an unbounded stack as auxiliary memory. Afterwards, Chomsky [19], Evey [20] and Schützenberger [21] showed that context-free grammars and PDA are equally expressive [22–24] [8, Section 8.2]. Inspired by PDA, we define the transitional semantics of grammars by labelled transition systems where states are stacks, labels encode the structure of sentences and transitions are small steps in the recursive derivation of sentences.

4.1. Stacks

Given a grammar $\mathcal{G} = \langle \mathscr{T},\ \mathscr{N},\ \overline{S},\ \mathscr{R}\rangle$, we let *stacks* $\varpi \in \mathcal{S} \triangleq (\mathscr{R}^{\bullet} \cup \mathscr{M})^\star$ be sentences over *rule states* $\mathscr{R}^{\bullet} \triangleq \{[A \to \sigma\textbf{.}\sigma'] \mid A \to \sigma\sigma' \in \mathscr{R}\}$ specifying the state of the derivation (σ has been derived while σ' is still to be derived) and markers $\mathscr{M} = \{\vdash, \dashv\}$ where $\vdash$ (resp. $\dashv$) marks the beginning (resp. the end) of a sentence. The *height* of a stack ϖ is its length $|\varpi|$.

Example 2. A stack ϖ for the grammar $A \to AA, A \to a$ is $\dashv[A \to AA\textbf{.}][A \to A\textbf{.}A][A \to a\textbf{.}]$. It records the ancestors in an infix traversal of a parse tree, as shown opposite. $\square$



4.2. Labels

We let $\mathscr{P} \triangleq \mathscr{O} \cup \mathscr{C}$ be the set of *parentheses* where $\mathscr{O} \triangleq \{(\!|A| \ A \in \mathscr{N}\}$ is the set of *opening parentheses* while $\mathscr{C} \triangleq \{A|\!) \mid A \in \mathscr{N}\}$ is the set of *closing parentheses*. We let *labels* $\ell \in \mathscr{L}$ be parentheses or terminals so that $\mathscr{L} \triangleq \mathscr{P} \cup \mathscr{T}$. A pair of parentheses $(\!|A\ldots A|\!)$ delimits the structure of a sentence deriving from nonterminal $A \in \mathscr{N}$ while terminals describe elements of the sentence.

4.3. Labelled transition system

Given a grammar $\mathcal{G} = \langle \mathscr{T},\ \mathscr{N},\ \overline{S},\ \mathscr{R}\rangle$, we define a *labelled transition system* $\mathsf{S}^t[\![\mathcal{G}]\!] \triangleq \langle \mathcal{S},\ \mathscr{L},\ \longrightarrow,\ \vdash\rangle$ where the initial state is $\vdash$ and the labelled transition relation $\xrightarrow{\ell}$, $\ell \in \mathscr{L}$ is

$$\vdash \xrightarrow{(\!|A} \dashv[A \to \textbf{.}\sigma], \qquad A \to \sigma \in \mathscr{R} \tag{1}$$

$$\varpi[A \to \sigma\textbf{.}a\sigma'] \xrightarrow{a} \varpi[A \to \sigma a\textbf{.}\sigma'], \qquad A \to \sigma a\sigma' \in \mathscr{R} \tag{2}$$

$$\varpi[A \to \sigma\textbf{.}B\sigma'] \xrightarrow{(\!|B} \varpi[A \to \sigma B\textbf{.}\sigma'][B \to \textbf{.}\varsigma], A \to \sigma B\sigma' \in \mathscr{R} \wedge B \to \varsigma \in \mathscr{R} \tag{3}$$

$$\varpi[A \to \sigma\textbf{.}] \xrightarrow{A|\!)} \varpi, \qquad A \to \sigma \in \mathscr{R}. \tag{4}$$

[2] An anonymous referee pointed out that this invention was preceded by [13–16], see [17,18].

Intuitively, the transition system $\mathsf{S}^t[\![\mathcal{G}]\!]$ generates the sentences of the language described by $\mathcal{G}$ by recursive infix traversal of their derivation tree using a stack to eliminate recursion. More precisely, (1) (resp. (3)) starts generating a terminal sentence for the nonterminal A (resp. B), (2) generates a terminal a, and (4) finishes the generation of a terminal sentence for the nonterminal A.

If we only want derivations from the grammar start symbol $\overline{S}$ then we replace transition rule (1) by

$$\vdash \xrightarrow{(\!|\overline{S}} \dashv[\overline{S} \to \mathbf{.}\sigma], \qquad \overline{S} \to \sigma \in \mathscr{R}. \tag{1'}$$

5. Maximal derivations

The *maximal derivation semantics* of a grammar is the set of all possible maximal derivations for this grammar where a *maximal derivation* is a finite labelled trace of maximal length generated by the transitional semantics.

Example 3. The maximal derivation for the sentence a of the grammar $\langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$ is $\vdash \xrightarrow{(\!|A} \dashv[A \to \mathbf{.}a] \xrightarrow{a} \dashv[A \to a\mathbf{.}] \xrightarrow{A|\!)} \dashv$ while for the sentence aa it is $\vdash \xrightarrow{(\!|A} \dashv[A \to \mathbf{.}AA] \xrightarrow{(\!|A} \dashv[A \to A\mathbf{.}A][A \to \mathbf{.}a] \xrightarrow{a} \dashv[A \to A\mathbf{.}A][A \to a\mathbf{.}] \xrightarrow{A|\!)} \dashv[A \to A\mathbf{.}A] \xrightarrow{(\!|A} \dashv[A \to AA\mathbf{.}][A \to \mathbf{.}a] \xrightarrow{a} \dashv[A \to AA\mathbf{.}][A \to a\mathbf{.}] \xrightarrow{A|\!)} \dashv[A \to AA\mathbf{.}] \xrightarrow{A|\!)} \dashv$. $\square$

5.1. Traces

Formally a *trace* $\theta \in \Theta[n]$ of length $|\theta| = n+1, n \geqslant 0$, has the form $\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n$ whence it is a pair $\theta = \langle\underline{\theta},\ \overline{\theta}\rangle$ where $\underline{\theta} \in [0, n] \mapsto \mathcal{S}$ is a nonempty finite sequence of stacks $\underline{\theta}_i = \varpi_n, i = 0, \ldots, n$ and $\overline{\theta} \in [0, n-1] \mapsto \mathscr{L}$ is a finite sequence of labels $\overline{\theta}_j = \ell_j, j = 0, \ldots, n-1$. Traces $\theta \in \Theta$ are nonempty, finite, of any length so $\Theta \triangleq \bigcup_{n\geqslant 0} \Theta[n]$.

Again concatenation is denoted by juxtaposition and extended to sets. We respectively identify a single state ϖ and a transition $\varpi \xrightarrow{\ell} \varpi'$ with the corresponding traces containing only the single state ϖ and the transition $\varpi \xrightarrow{\ell} \varpi'$. By abuse of notation, a trace $\varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n$ is also understood as the concatenation of $\varpi_0, \xrightarrow{\ell_0}, \varpi_1, \ldots, \varpi_{n-1}, \xrightarrow{\ell_{n-1}}, \varpi_n$ which, informally, *matches the trace pattern* $\varsigma_0\varpi_1 \ldots \varsigma_{n-1}\varpi_n\varsigma_n$ by letting $\varsigma_0 = \varpi_0 \xrightarrow{\ell_0}, \ldots, \varsigma_{n-1} = \varpi_{n-1} \xrightarrow{\ell_{n-1}}$ and $\varsigma_n = \epsilon$. We also need the *junction* of sets of traces, as follows

$$T \mathbin{⨾} T' \quad \triangleq \quad \{\theta \xrightarrow{\ell} \varpi \xrightarrow{\ell'} \theta' \mid \theta \xrightarrow{\ell} \varpi \in T \wedge \varpi' \xrightarrow{\ell'} \theta' \in T' \wedge \varpi = \varpi'\}.$$

The *selection* of the traces in T for nonterminal B is denoted $T.B$ defined as

$$T.B \quad \triangleq \quad \{\varpi \xrightarrow{(\!|B} \theta \mid \varpi \xrightarrow{(\!|B} \theta \in T\}.$$

For the recursive *incorporation* of a derivation $\vdash \xrightarrow{\ell_0} \dashv\varpi_1 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv$ into another one, we need the operation

$$\langle\varpi,\ \varpi'\rangle \uparrow \vdash \xrightarrow{\ell_0} \dashv\varpi_1 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv \quad \triangleq \quad \varpi \xrightarrow{\ell_0} \varpi'\varpi_1 \ldots \varpi'\varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi'$$
$$\langle\varpi,\ \varpi'\rangle \uparrow T \quad \triangleq \quad \{\langle\varpi,\ \varpi'\rangle \uparrow \tau \mid \tau \in T\}.$$

Example 4. We have $\langle\dashv[A \to \mathbf{.}AA],\ \dashv[A \to A\mathbf{.}A]\rangle \uparrow \vdash \xrightarrow{(\!|A} \dashv[A \to \mathbf{.}a] \xrightarrow{a} \dashv[A \to a\mathbf{.}] \xrightarrow{A|\!)} \dashv \;=\; \dashv[A \to \mathbf{.}AA] \xrightarrow{(\!|A} \dashv[A \to A\mathbf{.}A][A \to \mathbf{.}a] \xrightarrow{a} \dashv[A \to A\mathbf{.}A][A \to a\mathbf{.}] \xrightarrow{A|\!)} \dashv[A \to A\mathbf{.}A]$ which we can recognize as the replacement of the first A deriving into a in the derivation for the sentence aa in Example 3. $\square$

5.2. Prefix, suffix and maximal derivations

A *derivation* of grammar $\mathcal{G}$ is a trace $\varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n$, $n \geqslant 0$ generated by the transition system $\mathsf{S}^t[\![\mathcal{G}]\!]$ that is $\forall i \in [0, n-1] : \varpi_i \xrightarrow{\ell_i} \varpi_{i+1}$. A *prefix derivation* of grammar $\mathcal{G}$ is a derivation of grammar $\mathcal{G}$ starting with an initial state $\varpi_0 = \vdash$. A *suffix derivation* of grammar $\mathcal{G}$ is derivation of grammar $\mathcal{G}$ ending with an final state $\forall \varpi \in \mathcal{S} : \forall \ell \in \mathscr{L} : \neg(\varpi_n \xrightarrow{\ell} \varpi)$, so that $\varpi_n = \dashv$ by def. (1)–(4) of $\longrightarrow$. A *maximal derivation* of grammar $\mathcal{G}$ is both a prefix and a suffix derivation of the grammar $\mathcal{G}$.

5.3. The well-parenthesized structure of prefix and maximal derivations

Derivations are well-parenthesized so that the grammatical structure of sentences can be described by trees. Let us define the *parenthesis abstraction* α^p for a stack ϖ by $\alpha^p(\varpi\varpi') \triangleq \alpha^p(\varpi')\alpha^p(\varpi)$, $\alpha^p(\vdash) = \alpha^p(\dashv) = \epsilon$ and $\alpha^p([A \to \sigma\text{.}\sigma']) \triangleq A\rrparenthesis$, for a label, $\alpha^p(a) \triangleq \epsilon$ for all $a \in \mathscr{T}$, $\alpha^p(\llparenthesis A) \triangleq \llparenthesis A$ and $\alpha^p(A\rrparenthesis) \triangleq A\rrparenthesis$, and for a trace $\alpha^p(\varpi_0 \xrightarrow{\ell_0} \varpi_1 \xrightarrow{\ell_1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) \triangleq \alpha^p(\ell_0)\alpha^p(\ell_1)\ldots\alpha^p(\ell_{n-1})\alpha^p(\varpi_n)$.

Lemma 5. *For any prefix derivation θ of a grammar $\mathcal{G}$, $\alpha^p(\theta) \in \mathbb{D}_{\mathscr{P},\varnothing}$ is a pure Dyck language. A maximal derivation $\theta = \vdash \xrightarrow{\ell_0} \varpi_1 \xrightarrow{\ell_1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv$ of $\mathcal{G}$ is well-parenthesized in that $\alpha^p(\theta) = \alpha^p(\ell_0)\alpha^p(\ell_1)\ldots\alpha^p(\ell_{n-1}) \in \mathbb{D}_{\mathscr{P},\varnothing}$ is a pure Dyck language.* □

Proof sketch. The proof is by induction on the length of θ, where the basis is true for the prefix derivation reduced to the initial state $\vdash$ and the induction step is for a prefix derivation of the form $\theta = \vdash \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n$ where $\alpha^p(\vdash \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1})$ is well-parenthesized by induction hypothesis is handled by case analysis using the Definition (1)–(4) of the transition relation $\mathsf{S}^t[\![\mathcal{G}]\!]$. □

Corollary 6. *A maximal derivation $\theta = \vdash \xrightarrow{\ell_0} \varpi_1 \xrightarrow{\ell_1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv$ of $\mathcal{G}$ is well-parenthesized in that $\alpha^p(\theta) = \alpha^p(\ell_0)\alpha^p(\ell_1)\ldots\alpha^p(\ell_{n-1}) \in \mathbb{D}_{\mathscr{P},\varnothing}$ is a pure Dyck language.* □

Proof. A maximal derivation θ of $\mathcal{G}$ is a prefix derivation $\vdash \xrightarrow{\ell_0} \varpi_1 \xrightarrow{\ell_1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n$ which is also a suffix derivation so $\varpi_n = \dashv$. It follows by Lemma 5 that $\alpha^p(\theta) = \alpha^p(\ell_0)\alpha^p(\ell_1)\ldots\alpha^p(\ell_{n-1})\alpha^p(\dashv) = \alpha^p(\ell_0)\alpha^p(\ell_1)\ldots\alpha^p(\ell_{n-1})$ since $\alpha^p(\dashv) = \epsilon$. □

5.4. Well-parenthesized traces

Corollary 6 leads to the definition of the set $\Theta_{()} \subseteq \Theta$ of *well-parenthesized traces*

$$\Theta_{()} \triangleq \{\vdash \xrightarrow{\ell_0} \varpi_1 \xrightarrow{\ell_1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv \in \Theta \mid \alpha^p(\ell_0)\alpha^p(\ell_1)\ldots\alpha^p(\ell_{n-1}) \in \mathbb{D}_{\mathscr{P},\varnothing}\}.$$

6. Prefix derivation semantics

The *prefix derivation semantics* $\vec{\mathsf{S}^{\partial}}[\![\mathcal{G}]\!]$ of a grammar $\mathcal{G} = \langle\mathscr{T},\ \mathscr{N},\ \bar{S},\ \mathscr{R}\rangle$ is the set of all prefix derivations for the labelled transition system $\langle\mathcal{S},\ \mathscr{L},\ \longrightarrow,\ \vdash\rangle$, that is

$$\vec{\mathsf{S}^{\partial}}[\![\mathcal{G}]\!] \triangleq \{\varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n \mid n \geqslant 0 \wedge \varpi_0 = \vdash \wedge \forall i \in [0, n-1] : \varpi_i \xrightarrow{\ell_i} \varpi_{i+1}\}.$$

Lemma 7. *If the prefix derivation semantics $\vec{\mathsf{S}^{\partial}}[\![\mathcal{G}]\!]$ of a grammar $\mathcal{G} = \langle\mathscr{T},\ \mathscr{N},\ \bar{S},\ \mathscr{R}\rangle$ contains a prefix derivation $\theta_1\varpi\theta_2$ then*

- *either $\varpi = \vdash$ if and only if $\theta_1 = \epsilon$.*
- *or the stack ϖ has the form $\varpi = \dashv[A_1 \to \eta_1 A_2\text{.}\eta'_1][A_2 \to \eta_2 A_3\text{.}\eta'_2]\ldots[A_n \to \eta_n\text{.}\eta'_n]$ where $A_i \to \eta_i A_{i+1}\eta'_i \in \mathscr{R}$ and $A_n \to \eta_n\eta'_n \in \mathscr{R}$ are grammar rules and $\theta_1 = \vdash \xrightarrow{\llparenthesis A_1} \theta'_1$.*
- *Moreover if $\theta_1\varpi\theta_2 \in \vec{\mathsf{S}^{\partial}}[\![\mathcal{G}]\!]\text{.}A$ then necessarily $A_1 = A$.* □

Proof sketch. The proof is by induction on the position of the stack ϖ in the prefix derivation $\theta_1\varpi\theta_2$ distinguishing the first position, $\varpi = \vdash$, the second position $\theta_1\varpi\theta_2 = \vdash \xrightarrow{\llparenthesis A} \varpi\theta_2$ with $\varpi = \dashv[A \to \text{.}\sigma]$ and $A \to \sigma \in \mathscr{R}$, observing that if $\theta_1\varpi\theta_2 \in \vec{\mathsf{S}^{\partial}}[\![\mathcal{G}]\!]\text{.}A$ then $A_1 = A$ by definition of the trace selection $\bullet\text{.}A$, and for the induction step where the lemma holds up to position i and ϖ is in position $i+1$ that we have $\varpi_i \xrightarrow{\ell_i} \varpi$ where the lemma holds for ϖ_i by induction hypothesis so that the lemma follows from the Definition (1)–(4) of $\xrightarrow{\ell_i}$. □

It has been shown in the more general context of [25, Th. 11] that we have the following fixpoint characterization of the prefix derivation semantics

Theorem 8.

$$\vec{\mathsf{S}^{\partial}}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\subseteq}\, \vec{\mathsf{F}^{\partial}}[\![\mathcal{G}]\!] = \mathbf{gfp}^{\subseteq}\, \vec{\mathsf{F}^{\partial}}[\![\mathcal{G}]\!]$$

where $\vec{\mathsf{F}^{\partial}}[\![\mathcal{G}]\!] \in \wp(\Theta) \mapsto \wp(\Theta)$ is a complete $\cup$ and $\cap$ morphism defined as

$$\vec{\mathsf{F}^{\partial}}[\![\mathcal{G}]\!] \triangleq \boldsymbol{\lambda} X \bullet \{\vdash\} \cup X \fatsemi \longrightarrow . \quad \square$$

Proof. See [25, Th. 11]. □

7. Transitional maximal derivation semantics

The *maximal derivation semantics* $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] \in \wp(\Theta)$ of a grammar $\mathcal{G} = \langle \mathcal{T},\ \mathcal{N},\ \overline{S},\ \mathcal{R}\rangle$ is the set of maximal derivations for the labelled transition system $\mathsf{S}^t[\![\mathcal{G}]\!] \triangleq \langle \mathcal{S},\ \mathcal{L},\ \longrightarrow,\ \vdash\rangle$, that is the set of finite traces starting in an initial state $\vdash$, where each step is generated by the transition relation $\longrightarrow$ and terminating in a blocking state, with no possible successor.[3]

$$\begin{aligned}\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] \quad \triangleq \quad & \{\varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n \mid n > 0 \wedge \varpi_0 = \vdash \wedge \\ & \forall i \in [0, n-1] : \varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \wedge \forall \varpi \in \mathcal{S} : \forall \ell \in \mathcal{L} : \neg(\varpi_n \xrightarrow{\ell} \varpi)\}.\end{aligned} \tag{5}$$

Lemma 9. *A maximal derivation of the transition system* $\mathsf{S}^t[\![\mathcal{G}]\!]$ *has the form* $\vdash \xrightarrow{(\!|A} \dashv[A \to .\sigma] \xrightarrow{\ell_1} \dashv\varpi_2 \ldots \dashv\varpi_{n-1} \xrightarrow{A|\!)} \dashv$ *where* $\varpi_{n-1} \neq \epsilon$. $\square$

Proof. Observe that maximal derivations are traces $\varpi'_0 \xrightarrow{\ell_0} \varpi'_1 \ldots \varpi'_{n-1} \xrightarrow{\ell_{n-1}} \varpi'_n$ necessarily start with the initial state $\varpi'_0 = \vdash$. Then the only possible derivation is $\vdash \xrightarrow{(\!|A} \dashv[A \to .\sigma]$ for some $A \to \sigma \in \mathcal{R}$ so ϖ'_1 has the form $\dashv\varpi_1$. Then, by induction, all states $\varpi'_i = \dashv\varpi_i$ where ϖ_i is not empty do have a successor which, by definition of the transition relation has the same form $\varpi'_{i+1} = \dashv\varpi_{i+1}$. Since maximal derivations are finite and maximal traces, the derivation must end with $\varpi'_n = \dashv\varpi_n$ without a possible successor in the transition relation $\neg(\exists\varpi \in \mathcal{S} : \exists\ell \in \mathcal{L} : \dashv\varpi_n \xrightarrow{\ell} \varpi)$. The only possible one is $\varpi'_n = \dashv$.

By Lemma 5, $\alpha^p(\vdash \xrightarrow{(\!|A} \dashv[A \to .\sigma] \xrightarrow{\ell_1} \dashv\varpi_2 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv) = (\!|A\, \alpha^p(\ell_1) \ldots \alpha^p(\ell_{n-1})$ is well-parenthesized so necessarily $\alpha^p(\ell_{n-1}) = A|\!)$ proving that $\ell_{n-1} = A|\!)$.

Observe that $\varpi_{n-1} \neq \epsilon$ since otherwise $\dashv \xrightarrow{A|\!)} \dashv$ which, by definition of $\longrightarrow$, does not hold. $\square$

Let us define the *final traces* $\Theta^{\dashv} \triangleq \{\theta \xrightarrow{\ell} \varpi \in \Theta \mid \varpi = \dashv\}$, the final traces abstraction $\alpha^{\dashv} \triangleq \boldsymbol{\lambda} X \bullet \Theta^{\dashv} \cap X$ (so that $\langle\Theta,\ \subseteq\rangle \underset{\alpha^{\dashv}}{\overset{\gamma^{\dashv}}{\leftrightarrows}} \langle\Theta^{\dashv},\ \subseteq\rangle$ with $\gamma^{\dashv} \triangleq \boldsymbol{\lambda} Y \bullet Y \cup \Theta \setminus \Theta^{\dashv}$). As a corollary of Lemma 9, the maximal derivation semantics is an abstraction of the prefix derivation semantics, as follows

$$\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] \quad = \quad \alpha^{\dashv}(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!]) \quad = \quad \mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!] \cap \Theta^{\dashv}. \tag{6}$$

8. Bottom-up fixpoint maximal derivation semantics

The maximal derivation semantics (5) can be expressed in structural fixpoint form.

Example 10. For the grammar $\mathcal{G} = \langle\{a, b\},\ \{A\},\ A,\ \{A \to aA, A \to b\}\rangle$, we have $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\subseteq}\ \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ where

$$\begin{aligned}&\hat{\mathsf{F}}^{\hat{d}}(T) \quad \triangleq \quad \vdash \xrightarrow{(\!|A} \dashv[A \to .b] \xrightarrow{b} \dashv[A \to b.] \xrightarrow{A|\!)} \dashv \cup \\ &\vdash \xrightarrow{(\!|A} (\dashv[A \to .aA]) \xrightarrow{a} (\langle\dashv[A \to a.A],\ \dashv[A \to aA.]\rangle \uparrow T.A) \fatsemi (\dashv[A \to aA.]) \xrightarrow{A|\!)} \dashv .\end{aligned}$$

The first iterates of $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ from $\hat{\mathsf{F}}^{\hat{d}}_0 = \varnothing$ (as defined in Appendix A.1) are

$$\begin{aligned}\hat{\mathsf{F}}^{\hat{d}}_1 \quad &= \quad \{\vdash \xrightarrow{(\!|A} \dashv[A \to .b] \xrightarrow{b} \dashv[A \to b.] \xrightarrow{A|\!)} \dashv\} \\ \hat{\mathsf{F}}^{\hat{d}}_2 \quad &= \quad \{\vdash \xrightarrow{(\!|A} \dashv[A \to .b] \xrightarrow{b} \dashv[A \to b.] \xrightarrow{A|\!)} \dashv, \\ & \qquad \vdash \xrightarrow{(\!|A} \dashv[A \to .aA] \xrightarrow{a} \dashv[A \to a.A] \xrightarrow{(\!|A} \dashv[A \to aA.][A \to .b] \xrightarrow{b} \\ & \qquad\quad \dashv[A \to aA.][A \to b.] \xrightarrow{A|\!)} \dashv[A \to aA.] \xrightarrow{A|\!)} \dashv\} \\ \ldots \quad & \quad\ \ \ldots \\ \hat{\mathsf{F}}^{\hat{d}}_\omega \quad &= \quad \mathbf{lfp}^{\subseteq}\ \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]. \quad \square\end{aligned}$$

[3] It is also possible to consider infinite traces in the style of [25] to cope with infinitary languages.

8.1. Bottom-up set of traces transformer

More generally, let us define the set of traces bottom-up transformer $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \in \wp(\Theta) \mapsto \wp(\Theta)$ as

$$\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda}\, T \bullet \bigcup_{A \to \sigma \in \mathscr{R}} \vdash \xrightarrow{\llparenthesis A} \hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \boldsymbol{.}\sigma]T \xrightarrow{A \rrparenthesis} \dashv \tag{7}$$

where $\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}\sigma'] \in \wp(\Theta) \mapsto \wp(\Theta)$ is defined as

$$\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}a\sigma'] \quad \triangleq \quad \boldsymbol{\lambda}\, T \boldsymbol{\cdot} (\dashv[A \to \sigma\boldsymbol{.}a\sigma']) \xrightarrow{a} \hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma a\boldsymbol{.}\sigma']T \tag{8}$$

$$\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}B\sigma'] \quad \triangleq \quad \boldsymbol{\lambda}\, T \boldsymbol{\cdot} (\langle \dashv[A \to \sigma\boldsymbol{.}B\sigma'],\ \dashv[A \to \sigma B\boldsymbol{.}\sigma'] \rangle \uparrow T.B) \,\mathbin{;}\, \hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma B\boldsymbol{.}\sigma']T \tag{9}$$

$$\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}] \quad \triangleq \quad \boldsymbol{\lambda}\, T \boldsymbol{\cdot} (\dashv[A \to \sigma\boldsymbol{.}]). \tag{10}$$

Lemma 11. *For all* $[A \to \sigma\boldsymbol{.}\sigma'] \in \mathscr{R}$, $\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}\sigma']$, *is upper continuous.* □

Proof. By forthcoming Lemma 28, observing that $\boldsymbol{\lambda}\, T \bullet \vdash \xrightarrow{\llparenthesis A} T \xrightarrow{A \rrparenthesis} \dashv$, $\boldsymbol{\lambda}\, T \bullet (\dashv[A \to \sigma\boldsymbol{.}a\sigma']) \xrightarrow{a} T$, $\boldsymbol{\lambda}\, T \bullet T.B$, $\langle \dashv[A \to \sigma\boldsymbol{.}B\sigma'],\ \dashv[A \to \sigma B\boldsymbol{.}\sigma'] \rangle \uparrow T$, $\mathbin{;}$, and concatenation are continuous. □

Lemma 12. *If all traces in* $T \subseteq \Theta$ *are derivations of the transition system* $\mathsf{S}^t[\![\mathcal{G}]\!]$ *then all traces in* $\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}\sigma']T$ *are generated by the transition system* $\mathsf{S}^t[\![\mathcal{G}]\!]$, *start in state* $(\dashv[A \to \sigma\boldsymbol{.}\sigma'])$ *and end in state* $(\dashv[A \to \sigma\sigma'\boldsymbol{.}])$. □

Proof. The proof is by induction on the length of σ'.

For the base case $\sigma' = \epsilon$, the trace is $(\dashv[A \to \sigma\boldsymbol{.}])$ by (10), which is a correct state in $\mathcal{S}$, whence a trace generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$.

If $\sigma' = a\sigma''$, then (8) applies. By induction hypothesis, all traces θ in $\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma a\boldsymbol{.}\sigma'']T$ are generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$, start in state $(\dashv[A \to \sigma a\boldsymbol{.}\sigma''])$ and end in state $(\dashv[A \to \sigma a\sigma''\boldsymbol{.}])$. By (2), $(\dashv[A \to \sigma\boldsymbol{.}a\sigma'']) \xrightarrow{a} (\dashv[A \to \sigma a\boldsymbol{.}\sigma''])$ is valid transition of $\mathsf{S}^t[\![\mathcal{G}]\!]$ so the trace $(\dashv[A \to \sigma\boldsymbol{.}a\sigma'']) \xrightarrow{a} \theta$ is generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$, starts with $(\dashv[A \to \sigma\boldsymbol{.}a\sigma''])$ and ends in state $(\dashv[A \to \sigma a\sigma''\boldsymbol{.}])$.

Otherwise $\sigma' = B\sigma''$ and (9) applies. All traces in T are assumed to be derivations of the transition system $\mathsf{S}^t[\![\mathcal{G}]\!]$, whence so are those in the subset $T.B$. By Lemma 9, these traces have the form $\vdash \xrightarrow{\llparenthesis A} \dashv[A \to \boldsymbol{.}\sigma] \xrightarrow{\ell_1} \dashv\varpi_2 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv$. So all traces in $\langle \dashv[A \to \sigma\boldsymbol{.}B\sigma''],\ \dashv[A \to \sigma B\boldsymbol{.}\sigma''] \rangle \uparrow T.B$ have the form $(\dashv[A \to \sigma\boldsymbol{.}B\sigma'']) \xrightarrow{\llparenthesis A} (\dashv[A \to \sigma B\boldsymbol{.}\sigma''][A \to \boldsymbol{.}\sigma]) \xrightarrow{\ell_1} (\dashv[A \to \sigma B\boldsymbol{.}\sigma'']\varpi_2) \ldots (\dashv[A \to \sigma B\boldsymbol{.}\sigma'']\varpi_{n-1}) \xrightarrow{\ell_{n-1}} (\dashv[A \to \sigma B\boldsymbol{.}\sigma''])$. These traces start with $(\dashv[A \to \sigma\boldsymbol{.}B\sigma''])$ and are generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$ since the first transition corresponds to (3) while, for the following ones, if $\varpi \xrightarrow{\ell} \varpi'$ is one of the transitions (2), (3) or (4) of $\mathsf{S}^t[\![\mathcal{G}]\!]$ then so is $\varpi''\varpi \xrightarrow{\ell} \varpi''\varpi'$. By induction hypothesis, all traces in $\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma B\boldsymbol{.}\sigma'']T$ are generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$, start with state $(\dashv[A \to \sigma B\boldsymbol{.}\sigma''])$ and end with state $(\dashv[A \to \sigma B\sigma''\boldsymbol{.}])$. It follows that the junction, whence by (9), that $\hat{\underset{\cdot}{\mathsf{F}}}^{\hat{d}}[A \to \sigma\boldsymbol{.}B\sigma'']$ starts with $(\dashv[A \to \sigma\boldsymbol{.}B\sigma''])$, is generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$ and ends with $(\dashv[A \to \sigma B\sigma''\boldsymbol{.}])$. □

Corollary 13. *If all traces in* T *are derivations of the transition system* $\mathsf{S}^t[\![\mathcal{G}]\!]$ *then so are all traces in* $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]T$. □

Proof. By (7), all traces in $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]T$ have the form $\vdash \xrightarrow{\llparenthesis A} \theta \xrightarrow{A \rrparenthesis} \dashv$ where θ is a trace of $\hat{\mathsf{F}}^{\hat{d}}(\dashv[A \to \boldsymbol{.}\sigma])T$. By Lemma 12, θ is generated by the transition system $\mathsf{S}^t[\![\mathcal{G}]\!]$, starts in state $(\dashv[A \to \boldsymbol{.}\sigma])$ and ends in state $(\dashv[A \to \sigma\boldsymbol{.}])$. But $\vdash \xrightarrow{\llparenthesis A} (\dashv[A \to \boldsymbol{.}\sigma])$ is a valid transition by (1) and $(\dashv[A \to \sigma\boldsymbol{.}]) \xrightarrow{A \rrparenthesis} \dashv$ is a valid transition of $\mathsf{S}^t[\![\mathcal{G}]\!]$ by (4) so $\vdash \xrightarrow{\llparenthesis A} \theta \xrightarrow{A \rrparenthesis} \dashv$ is generated by $\mathsf{S}^t[\![\mathcal{G}]\!]$. Since it ends by state $\dashv$ without successor, it is also maximal whence a maximal derivation of $\mathsf{S}^t[\![\mathcal{G}]\!]$. □

8.2. Bottom-up fixpoint maximal derivation semantics

The derivation semantics of a grammar $\mathcal{G}$ can be expressed in fixpoint form for transformer $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ as follows

Theorem 14. $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] = \boldsymbol{\mathit{lfp}}^{\subseteq}\, \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$. □

Proof. (a) Because $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ is continuous (indeed it preserves existing lubs), we have $\mathbf{lfp}^{\subseteq}\, \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] = T^{\omega} \triangleq \bigcup_{i \geqslant 0} T^i$ where the iterates (as defined in Appendix A.1) are $T^0 \triangleq \varnothing$, $T^{n+1} \triangleq \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!](T^n)$.

(b) All traces in $T^0 = \varnothing$, whence by recurrence using Corollary 13, all traces in the T^i, hence all those in $T^{\omega} = \mathbf{lfp}^{\subseteq}\, \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ are derivations of the transition system $\mathsf{S}^t[\![\mathcal{G}]\!]$ so $\mathbf{lfp}^{\subseteq}\, \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \subseteq \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!]$.

(c) Reciprocally, let θ be a derivation of $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!]$. By Lemma 9, θ is of the form $\vdash \xrightarrow{(\!|A} \dashv[A \to \centerdot\sigma] \xrightarrow{\ell_1} \dashv\varpi_2 \ldots \dashv\varpi_{n-1} \xrightarrow{A|\!)} \dashv$ where $\varpi_{n-1} \neq \epsilon$. We must prove that θ is in $\mathbf{lfp}^{\subseteq}\, \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ that is in some T^i, $i > 0$. The proof is by recurrence on the maximal height $h = \max\{|\dashv[A \to \centerdot\sigma]|, |\dashv\varpi_2|, \ldots, |\dashv\varpi_{n-1}|\} \geqslant 2$ of the stacks in θ.

By Definition 7 of $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]T = \bigcup_{A\to\sigma\in\mathscr{R}} \vdash \xrightarrow{(\!|A} \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \centerdot\sigma]T \xrightarrow{A|\!)} \dashv$, it is sufficient to prove that $\theta' = \dashv[A \to \centerdot\sigma] \xrightarrow{\ell_1} \dashv\varpi_2 \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \centerdot\sigma]T^i$ for some $i > 0$.

If σ is empty then, by Definition 4, θ' is reduced to $\dashv[A \to \centerdot]$, which by (10) belongs to $\hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \centerdot]T^i$ for all $i \geqslant 0$.

Otherwise, σ is not empty.

For the base case $h = 2$, rule (3) would yield to a maximum stack height of at least 3. Hence this rule is not useable for trace θ'. This means that σ may only contain terminals so that the trace can be built using rules (2) and (4) only. By induction on the length $|\sigma|$ of σ, the trace will be in $\hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \centerdot\sigma]T^0$ where $T^0 = \varnothing$ using respectively (8) and (10).

For the inductive case $h > 2$, we solve the more general problem of proving, given $\sigma = \sigma'\sigma''$, that $\dashv[A \to \sigma'\centerdot\sigma''] \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'\centerdot\sigma'']T^i$ for some $i > 0$ where $\varpi_{n-1} \neq \epsilon$. We can then conclude by choosing $\sigma' = \epsilon$ and $\sigma'' = \sigma$. The proof proceeds by induction on the length $|\sigma''|$ of σ'' and there are three cases.

— In case $\sigma'' = \epsilon$, then by (4), we must prove that $\dashv[A \to \sigma'\centerdot] \xrightarrow{A|\!)} \dashv\varpi_{k+1} \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'\centerdot]T^i$ for some $i > 0$ where $\varpi_{k+1} = \epsilon$. Because $\dashv$ has no successor by $\longrightarrow$, we have $k + 1 = n - 1$ but then $\varpi_{n-1} = \epsilon$, in contradiction with our assumption. So this case is impossible.

— In case $\sigma'' = a\sigma'''$, $\dashv[A \to \sigma'\centerdot a\sigma'''] \xrightarrow{\ell_k} \dashv\varpi_{k+1}$ must be of the form $\dashv[A \to \sigma'\centerdot a\sigma'''] \xrightarrow{a} \dashv[A \to \sigma'a\centerdot\sigma''']$ by (2) so that $\ell_k = a$ and $\varpi_{k+1} = [A \to \sigma'a\centerdot\sigma''']$. Since $|\sigma'''| < |\sigma''|$ there exists, by induction hypothesis, some $i \geqslant 0$ such that $\dashv[A \to \sigma'a\centerdot\sigma'''] \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'a\centerdot\sigma''']T^i$ so that we conclude that $\dashv[A \to \sigma'\centerdot a\sigma'''] \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'\centerdot a\sigma''']T^i$ by (8).

— In case $\sigma'' = B\sigma'''$, $\dashv[A \to \sigma'\centerdot B\sigma'''] \xrightarrow{\ell_k} \dashv\varpi_{k+1}$ must be of the form $\dashv[A \to \sigma'\centerdot B\sigma'''] \xrightarrow{(\!|B} \dashv[A \to \sigma'B\centerdot\sigma'''][B \to \centerdot\varsigma]$ where $B \to \varsigma \in \mathscr{R}$ by (3) so that $\ell_k = (\!|B$ and $\varpi_{k+1} = [A \to \sigma'B\centerdot\sigma'''][B \to \centerdot\varsigma]$. By Lemma 5, $\alpha^p(\theta) = \alpha^p(\vdash \xrightarrow{(\!|A} \dashv[A \to \centerdot\sigma] \xrightarrow{\ell_1} \dashv\varpi_2 \ldots \dashv[A \to \sigma'\centerdot B\sigma'''] \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_{n-1} \xrightarrow{A|\!)} \dashv)$ is well-parenthesized so that the opening parenthesis $(\!|B$ in ℓ_k must have a matching closing parenthesis $B|\!)$ in ℓ_m where $k < m \leqslant n-1$. By definition of $\longrightarrow$ and (4), we must have $\dashv\varpi_m = \varpi[B \to \varsigma\centerdot] \xrightarrow{B|\!)} \varpi = \dashv\varpi_{m+1}$. Moreover $m \neq n-1$ since θ' excludes the pair of external parentheses in θ.

Observe that in θ', (1) is not applicable so that the only two transitions that can change the stack height in θ' are (3) and (4). The stack height is increased by one in (3) on opening parentheses and decreased by one in (4) for closing parentheses. Since θ' is well-parenthesized, it follows that the stack have the same height on matching parentheses. Moreover the transitions in $\mathsf{S}^t[\![\mathcal{G}]\!]$ never change the bottom of the stack. Since $\dashv\varpi_k = \dashv[A \to \sigma'\centerdot B\sigma''']$, $\dashv\varpi_{k+1} = \dashv[A \to \sigma'B\centerdot\sigma'''][B \to \centerdot\varsigma]$ the stack around the matching parentheses are $\dashv\varpi_m = \varpi[B \to \varsigma\centerdot] = \dashv[A \to \sigma'B\centerdot\sigma'''][B \to \varsigma\centerdot]$ and $\varpi = \dashv\varpi_{m+1} = \dashv[A \to \sigma'B\centerdot\sigma''']$. Moreover the bottom of the stack in between is $\dashv[A \to \sigma'B\centerdot\sigma''']$. It follows that we can rewrite $\dashv\varpi_k \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_m \xrightarrow{\ell_m} \dashv\varpi_{m+1}$ in the form $\langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \dashv[A \to \sigma'B\centerdot\sigma''']\rangle \uparrow \varpi'_k \xrightarrow{\ell_k} \varpi'_{k+1}\dashv\varpi'_m \xrightarrow{\ell_m} \varpi'_{m+1}$ where $\theta'' = \varpi'_k \xrightarrow{\ell_k} \varpi'_{k+1} \ldots \varpi'_m \xrightarrow{\ell_m} \varpi'_{m+1} = \vdash \xrightarrow{(\!|B} \dashv[B \to \varsigma\centerdot] \ldots \dashv[B \to \centerdot\varsigma] \xrightarrow{B|\!)} \dashv$.

Since the maximal height of the stacks in θ'' are strictly less than that in θ', there exists $i \geqslant 0$ such that $\theta'' \in T^i$, whence by definition of selection $\theta'' \in T^i.B$ since θ'' starts with label $(\!|B$. It follows that $\dashv\varpi_k \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_m \xrightarrow{\ell_m} \dashv\varpi_{m+1} = \langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \theta''\rangle \uparrow \in \langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \dashv[A \to \sigma'B\centerdot\sigma''']\rangle \uparrow T^i.B$. Since the fixpoint iterates are $\subseteq$-increasing and $\langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \dashv[A \to \sigma'B\centerdot\sigma''']\rangle \uparrow \bullet$ is monotone, we also have $\dashv\varpi_k \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_m \xrightarrow{\ell_m} \dashv\varpi_{m+1} \in \langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \dashv[A \to \sigma'B\centerdot\sigma''']\rangle \uparrow T^p.B$ for all $p \geqslant i$.

Since $|\sigma'''| < |\sigma''|$ there exists, by induction hypothesis, some $i \geqslant 0$ such that $\dashv\varpi_{m+1} \ldots \dashv\varpi_{n-1} = \dashv[A \to \sigma'B\centerdot\sigma'''] \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'B\centerdot\sigma''']T^j$. Since the fixpoint iterates are $\subseteq$-increasing and $\hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'B\centerdot\sigma''']$ is monotone, we also have $\dashv\varpi_{m+1} \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'B\centerdot\sigma''']T^p$ for all $p \geqslant j$.

If we let $p = \max(i, j)$, we have $\dashv\varpi_k \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_m \xrightarrow{\ell_m} \dashv\varpi_{m+1} \in \langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \dashv[A \to \sigma'B\centerdot\sigma''']\rangle \uparrow T^p.B$ and $\dashv\varpi_{m+1} \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'B\centerdot\sigma''']T^p$ so by (9), $\dashv\varpi_k \xrightarrow{\ell_k} \dashv\varpi_{k+1} \ldots \dashv\varpi_m \xrightarrow{\ell_m} \dashv\varpi_{m+1} \ldots \dashv\varpi_{n-1} \in \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma\centerdot B\sigma']T^p = (\langle\dashv[A \to \sigma'\centerdot B\sigma'''],\ \dashv[A \to \sigma'B\centerdot\sigma''']\rangle \uparrow T^p.B) \,\mathring{\circ}\, \hat{\mathsf{F}}^{\hat{d}}_{\centerdot}[A \to \sigma'B\centerdot\sigma''']T^p$, as required. □

The fixpoint structural big-step maximal derivation semantics of a context-free grammar $\mathcal{G}$ in Theorem 14 is "bottom-up" in that when abstracting to derivation or syntax, these trees are constructed bottom-up (and left to right) which corresponds to the construction of traces by induction on their length, that is smaller ones first (and left to right).

9. Protoderivations

Prototraces (formally defined below) are traces in construction containing nonterminal variables which are placeholders for unknown prototraces to be substituted for the nonterminal variables. Protoderivations are prototraces generated by the grammar, initially a nonterminal variable (such as the grammar axiom), obtained by top-down replacement of a nonterminal on the left-hand side of a grammar rule by the corresponding right-hand side, until no nonterminal variable is left.

9.1. Examples of protoderivations

Example 15. A prototrace derivation for the grammar $\mathcal{G} = \langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$ is (the prototrace derivation relation is written $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$)

$$\begin{aligned}
& \vdash \xrightarrow{\boxed{A}} \dashv \\
\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}\ & \vdash \xrightarrow{(\!|A} \dashv[A \to .AA] \xrightarrow{\boxed{A}} \dashv[A \to A.A] \xrightarrow{\boxed{A}} \dashv[A \to AA.] \xrightarrow{A|\!)} \dashv \\
\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}\ & \vdash \xrightarrow{(\!|A} \dashv[A \to .AA] \xrightarrow{\boxed{A}} \dashv[A \to A.A] \xrightarrow{(\!|A} \dashv[A \to AA.][A \to .a] \xrightarrow{a} \dashv[A \to AA.][A \to a.] \xrightarrow{A|\!)} \dashv[A \to AA.] \xrightarrow{A|\!)} \dashv \\
\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}\ & \vdash \xrightarrow{(\!|A} \dashv[A \to .AA] \xrightarrow{(\!|A} \dashv[A \to A.A][A \to .a] \xrightarrow{a} \dashv[A \to A.A][A \to a.] \xrightarrow{A|\!)} \dashv[A \to A.A] \xrightarrow{(\!|A} \dashv[A \to AA.][A \to .a] \xrightarrow{a} \dashv[A \to AA.][A \to a.] \xrightarrow{A|\!)} \dashv[A \to AA.] \xrightarrow{A|\!)} \dashv. \quad \square
\end{aligned}$$

9.2. Prototraces

To each nonterminal $A \in \mathcal{N}$ we associate a *nonterminal variable* $\boxed{A}$ representing an unknown prototrace for A. The set of nonterminal variables is $\mathcal{N}^{\square} \triangleq \{\boxed{A} \mid A \in \mathcal{N}\}$.

A *prototrace* $\pi \in \Pi^n$ of length $|\pi| = n + 1$, $n \geqslant 0$, has the form $\pi = \varpi_0 \xrightarrow{\kappa_0} \varpi_1 \ldots \varpi_{n-1} \xrightarrow{\kappa_{n-1}} \varpi_n$ whence is a pair $\pi = \langle \underline{\pi},\ \overline{\pi}\rangle$ where $\underline{\pi} \in [0, n] \mapsto \mathcal{S}$ is a nonempty finite sequence of stacks $\underline{\pi}_i = \varpi_n$, $i = 0, \ldots, n$ and $\overline{\pi} \in [0, n-1] \mapsto (\mathcal{L} \cup \mathcal{N}^{\square})$ is a finite sequence of labels or nonterminal variables $\overline{\pi}_j = \kappa_j$, $j = 0, \ldots, n-1$. Prototraces $\pi \in \Pi$ are nonempty, finite, of any length so $\Pi \triangleq \bigcup_{n\geqslant 0} \Pi^n$ and $\Theta \subseteq \Pi$.

Again prototrace pattern matching, prototrace concatenation, set of prototraces concatenation, the assimilation of a single state ϖ and a transition $\varpi \xrightarrow{\ell} \varpi'$ with the corresponding prototraces, the junction $\fatsemi$ of sets of prototraces, the selection $P.B$ of the prototraces in P for nonterminal B and the stack incorporation in a prototrace $\langle \varpi,\ \varpi'\rangle \uparrow \pi$ or a set T of prototraces $\langle \varpi,\ \varpi'\rangle \uparrow T$ are defined as for traces and sets of traces.

9.3. Prototrace generated by a grammar rule

The *prototrace generated by a grammar rule* $A \to \sigma \in \mathcal{R}$ is $\check{\mathsf{R}}^{\check{D}}[A \to \sigma]$ where $\check{\mathsf{R}}^{\check{D}} \in \mathcal{R} \mapsto \Pi$ is

$$\check{\mathsf{R}}^{\check{D}}[A \to \sigma] \triangleq \vdash \xrightarrow{(\!|A} \check{\mathsf{R}}^{\check{D}}_{.}[A \to .\sigma] \xrightarrow{A|\!)} \dashv \tag{11}$$

$$\check{\mathsf{R}}^{\check{D}}_{.}[A \to \sigma.a\sigma'] \triangleq \dashv[A \to \sigma.a\sigma'] \xrightarrow{a} \check{\mathsf{R}}^{\check{D}}_{.}[A \to \sigma a.\sigma'] \tag{12}$$

$$\check{\mathsf{R}}^{\check{D}}_{.}[A \to \sigma.B\sigma'] \triangleq \dashv[A \to \sigma.B\sigma'] \xrightarrow{\boxed{B}} \check{\mathsf{R}}^{\check{D}}_{.}[A \to \sigma B.\sigma'] \tag{13}$$

$$\check{\mathsf{R}}^{\check{D}}_{.}[A \to \sigma.] \triangleq \dashv[A \to \sigma.]. \tag{14}$$

Example 16. For the grammar $\mathcal{G} = \langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$, the prototrace generated for the grammar rules $A \to a$ and $A \to AA$ is respectively

$$\begin{aligned}
\check{\mathsf{R}}^{\check{D}}[A \to a] &= \vdash \xrightarrow{(\!|A} \dashv[A \to .a] \xrightarrow{a} \dashv[A \to a.] \xrightarrow{A|\!)} \dashv, \text{ and} \\
\check{\mathsf{R}}^{\check{D}}[A \to AA] &= \vdash \xrightarrow{(\!|A} \dashv[A \to .AA] \xrightarrow{\boxed{A}} \dashv[A \to A.A] \xrightarrow{\boxed{A}} \dashv[A \to AA.] \xrightarrow{A|\!)} \dashv. \quad \square
\end{aligned}$$

9.4. Prototrace derivation

The *prototrace derivation* relation $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}} \in \wp(\Pi \times \Pi)$ for a grammar $\mathcal{G} = \langle \mathcal{T},\ \mathcal{N},\ \overline{S},\ \mathcal{R}\rangle$ ($\boxed{D}\!\!\Longrightarrow$ when $\mathcal{G}$ is understood) consists in replacing one or several nonterminal variables by the prototrace generated by a grammar rule for that nonterminal.

Formally, the prototrace derivation $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}} \in \wp(\Pi \times \Pi)$ is defined as follows

$$\begin{aligned} & \pi \ \boxed{D}\!\!\Longrightarrow_{\mathcal{G}} \pi' \\ \triangleq\ & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, \varpi_1, \ldots, \varpi_{n+1} \in \mathcal{S}, A_1, \ldots, A_n \in \mathcal{N}, \sigma_1, \ldots, \sigma_n \in \mathcal{V}^\star : \\ & \pi = \varsigma_1 \varpi_1 \xrightarrow{\boxed{A_1}} \varpi_2 \varsigma_2 \ldots \varsigma_n \varpi_n \xrightarrow{\boxed{A_n}} \varpi_{n+1} \varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathcal{R} \wedge \\ & \pi' = \varsigma_1 \langle \varpi_1,\ \varpi_2 \rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A_1 \to \sigma_1] \varsigma_2 \ldots \varsigma_n \langle \varpi_n,\ \varpi_{n+1} \rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A_n \to \sigma_n] \varsigma_{n+1}. \end{aligned} \tag{15}$$

10. Transitional maximal protoderivation semantics

The *top-down maximal protoderivation semantics* $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \wp(\Pi)$ of a context-free grammar $\mathcal{G}$ is defined using the prototrace derivation transition relation $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$ as

$$\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda} A \bullet \{\pi \in \Pi \mid (\vdash \xrightarrow{\boxed{A}} \dashv)\ \boxed{D}\!\!\Longrightarrow^{\star}_{\mathcal{G}} \pi\} \tag{16}$$

where r^n, $n \in \mathbb{N}$ are the powers of relation r, $r^{n\star} \triangleq \bigcup_{i<n} r^i$ (so that $r^{0\star} \triangleq \bigcup \varnothing = \varnothing$), r^+ (resp. $r^\star$) is the transitive closure (resp. reflexive transitive closure) of r.

The protoderivation semantics $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]$ is "top-down" in that it starts from the grammar nonterminal variable $\boxed{A}$, $A \in \mathcal{N}$ and expands the nonterminal variables into their derivations until reaching a terminal derivation without nonterminal variables. When abstracting to protoderivation or protosyntax trees, these trees are constructed from the root towards the terminal leaves.

11. Top-down fixpoint maximal protoderivation semantics

The top-down maximal protoderivation semantics of a context-free grammar $\mathcal{G}$ can be expressed in fixpoint form, as follows (where post $\in \wp(\Sigma) \mapsto \wp(\Sigma)$ is $\text{post}[r]X \triangleq \{s' \in \Sigma \mid \exists s \in X : \langle s,\ s' \rangle \in r\}$)

Theorem 17. $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!] = \textbf{\textit{lfp}}^{\dot{\subseteq}}\ \check{\mathsf{F}}^{\check{D}}[\![\mathcal{G}]\!]$ *where* $\dot{\subseteq}$ *is the pointwise extension of* $\subseteq$ *and the set of prototraces transformer* $\check{\mathsf{F}}^{\check{D}}[\![\mathcal{G}]\!] \in (\mathcal{N} \mapsto \wp(\Pi)) \mapsto (\mathcal{N} \mapsto \wp(\Pi))$ *is*

$$\check{\mathsf{F}}^{\check{D}}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda} \phi \bullet \boldsymbol{\lambda} A \bullet \{\vdash \xrightarrow{\boxed{A}} \dashv\} \cup \text{post}[\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}]\phi(A). \quad \square$$

Proof. By [26, Th. 10-4.3] since $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!](A)$ is the set of reachable states for $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$ from the singleton $\{\vdash \xrightarrow{\boxed{A}} \dashv\}$. $\square$

Example 18. For the example grammar $\mathcal{G} = \langle \{a, b\},\ \{A\},\ A,\ \{A \to aA, A \to b\} \rangle$, we have

$$\begin{aligned} \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}b] &= \vdash \xrightarrow{(\!|A} \dashv[A \to \boldsymbol{.}b] \xrightarrow{b} \dashv[A \to b\boldsymbol{.}] \xrightarrow{A|\!)} \dashv \\ \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}aA] &= \vdash \xrightarrow{(\!|A} \dashv[A \to \boldsymbol{.}aA] \xrightarrow{a} \dashv[A \to a\boldsymbol{.}A] \xrightarrow{\boxed{A}} \dashv[A \to aA\boldsymbol{.}] \xrightarrow{A|\!)} \dashv \end{aligned}$$

the first few iterates of $\check{\mathsf{F}}^{\check{D}}[\![\mathcal{G}]\!]$ (as defined in Appendix A.1) are

$$\begin{aligned} \check{\mathsf{F}}^{\check{D}}_0 &= \varnothing \\ \check{\mathsf{F}}^{\check{D}}_1 &= \{\vdash \xrightarrow{\boxed{A}} \dashv\} \\ \check{\mathsf{F}}^{\check{D}}_2 &= \{\vdash \xrightarrow{\boxed{A}} \dashv,\quad \langle \vdash,\ \dashv \rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}b],\quad \langle \vdash,\ \dashv \rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}aA]\} \\ &= \{\vdash \xrightarrow{\boxed{A}} \dashv,\quad \vdash \xrightarrow{(\!|A} \dashv[A \to \boldsymbol{.}b] \xrightarrow{b} \dashv[A \to b\boldsymbol{.}] \xrightarrow{A|\!)} \dashv, \\ &\quad \vdash \xrightarrow{(\!|A} \dashv[A \to \boldsymbol{.}aA] \xrightarrow{a} \dashv[A \to a\boldsymbol{.}A] \xrightarrow{\boxed{A}} \dashv[A \to aA\boldsymbol{.}] \xrightarrow{A|\!)} \dashv\} \\ \check{\mathsf{F}}^{\check{D}}_3 &= \{\vdash \xrightarrow{\boxed{A}} \dashv,\quad \langle \vdash,\ \dashv \rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}b],\quad \langle \vdash,\ \dashv \rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}aA], \\ &\quad \vdash \xrightarrow{(\!|A} \dashv[A \to \boldsymbol{.}aA] \xrightarrow{a} \langle \dashv[A \to a\boldsymbol{.}A],\ \dashv[A \to aA\boldsymbol{.}] \rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}b] \xrightarrow{A|\!)} \dashv, \\ &\quad \vdash \xrightarrow{(\!|A} \dashv[A \to \boldsymbol{.}aA] \xrightarrow{a} \langle \dashv[A \to a\boldsymbol{.}A],\ \dashv[A \to aA\boldsymbol{.}] \rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \boldsymbol{.}aA] \xrightarrow{A|\!)} \dashv\} \end{aligned}$$

$=\{\vdash \xrightarrow{\boxed{A}} \dashv,\quad \vdash \xrightarrow{(\!|A} \dashv[A \to \bullet b] \xrightarrow{b} \dashv[A \to b\bullet] \xrightarrow{A|\!)} \dashv,$
$\vdash \xrightarrow{(\!|A} \dashv[A \to \bullet aA] \xrightarrow{a} \dashv[A \to a\bullet A] \xrightarrow{\boxed{A}} \dashv[A \to aA\bullet] \xrightarrow{A|\!)} \dashv,$
$\vdash \xrightarrow{(\!|A} \dashv[A \to \bullet aA] \xrightarrow{a} \dashv[A \to a\bullet A] \xrightarrow{(\!|A} \dashv[A \to aA\bullet][A \to \bullet b] \xrightarrow{b} \dashv[A \to aA\bullet][A \to b\bullet] \xrightarrow{A|\!)} \dashv[A \to aA\bullet] \xrightarrow{A|\!)} \dashv,$
$\vdash \xrightarrow{(\!|A} \dashv[A \to \bullet aA] \xrightarrow{a} \dashv[A \to a\bullet A] \xrightarrow{(\!|A} \dashv[A \to aA\bullet][A \to \bullet aA] \xrightarrow{a} \dashv[A \to aA\bullet][A \to a\bullet A] \xrightarrow{\boxed{A}} \dashv[A \to aA\bullet][A \to aA\bullet] \xrightarrow{A|\!)} \dashv[A \to aA\bullet] \xrightarrow{A|\!)} \dashv\}$

etc. □

12. Abstraction of the top-down protoderivation semantics into the bottom-up derivation semantics

12.1. Characterization of the maximal derivation semantics by prototrace derivation

The trace derivations $\theta \in \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].A$ for a nonterminal A can be constructed top-down using the prototrace derivation $\overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}$ as $(\vdash \xrightarrow{\boxed{A}} \dashv) \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \theta$.

Lemma 19. *If* $T = \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$ *then* $\hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma\bullet\sigma'](T) = \{\pi \in \Theta \mid \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$. □

Proof. By induction on the length $|\sigma'|$ of σ'. There are three cases.

— $\hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma\bullet a\sigma'](T)$

$= \{(\dashv[A \to \sigma\bullet a\sigma']) \xrightarrow{a} \pi \in \Theta \mid \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma a\bullet\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi \wedge \pi \in \Theta\}$ ≀def. (8), ind. hyp., and def. concatenation≀

$= \{\pi \in \Theta \mid \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet a\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$ ≀def. (15) of $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$ and $\overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}$, def. (12) of $\check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet a\sigma']$≀

— $\hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma\bullet B\sigma'](T)$

$= (\langle\dashv[A \to \sigma\bullet B\sigma'],\ \dashv[A \to \sigma B\bullet\sigma']\rangle \uparrow \{\pi \in \Theta \mid \exists A' \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A'}} \dashv) \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}.B) \,\mathring{,}\, \hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma B\bullet\sigma'](T)$
≀def. (9) of $\hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma\bullet B\sigma']$ and def. T≀

$= \bigcup\{\langle\dashv[A \to \sigma\bullet B\sigma'],\ \dashv[A \to \sigma B\bullet\sigma']\rangle \uparrow \pi \,\mathring{,}\, \hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma B\bullet\sigma'](T) \mid \pi \in \Theta \wedge (\vdash \xrightarrow{\boxed{B}} \dashv) \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$
≀def. $\langle\bullet,\ \bullet\rangle \uparrow \bullet$, $\Theta.B$, (15) of $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$ and $\overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}$ so that necessarily $A' = B$≀

$= \bigcup\{\langle\dashv[A \to \sigma\bullet B\sigma'],\ \dashv[A \to \sigma B\bullet\sigma']\rangle \uparrow \pi \,\mathring{,}\, \{\pi' \in \Theta \mid \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma B\bullet\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi'\} \mid \pi \in \Theta \wedge (\vdash \xrightarrow{\boxed{B}} \dashv) \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$
≀ind. hyp.≀

$= \{\pi'' \,\mathring{,}\, \pi' \mid \pi'' \in \Theta \wedge (\dashv[A \to \sigma\bullet B\sigma'] \xrightarrow{\boxed{B}} \dashv[A \to \sigma B\bullet\sigma']) \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi'' \wedge \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma B\bullet\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi' \wedge \pi' \in \Theta\}$
≀def. $\mathring{,}$, (15) of $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$, $\overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}$ and $\pi'' = \langle\dashv[A \to \sigma\bullet B\sigma'],\ \dashv[A \to \sigma B\bullet\sigma']\rangle \uparrow \pi$≀

$= \{\pi \in \Theta \mid \dashv[A \to \sigma\bullet B\sigma'] \xrightarrow{\boxed{B}} \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma B\bullet\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$
≀def. (15) of $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$, $\overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}$ and $\pi = \pi'' \,\mathring{,}\, \pi'$, def. $\mathring{,}$ and $\check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma B\bullet\sigma']$ which starts with $\dashv[A \to \sigma B\bullet\sigma']$≀

$= \{\pi \in \Theta \mid \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet B\sigma'] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$ ≀def. (13) of $\check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet B\sigma']$≀

— $\hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma\bullet](T)$

$= \{\pi \in \Theta \mid \dashv[A \to \sigma\bullet] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$
≀def. (10) of $\hat{\mathsf{F}}^{\hat{d}}_{\cdot}[A \to \sigma\bullet]$ and def. (15) of $\boxed{D}\!\!\Longrightarrow_{\mathcal{G}}$ and $\overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}$ so that necessarily $\pi = \dashv[A \to \sigma\bullet]$ since $\dashv[A \to \sigma\bullet]$ contains no nonterminal variable≀

$= \{\pi \in \Theta \mid \check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet] \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$ ≀def. (14) of $\check{\mathsf{R}}^{\check{D}}_{\cdot}[A \to \sigma\bullet]$≀. □

Lemma 20. *Let $\hat{\mathsf{F}}^{\hat{d}}_n$ be the iterates of $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ from $\hat{\mathsf{F}}^{\hat{d}}_0 = \varnothing$ (as defined in Appendix A.1). We have*

$$\hat{\mathsf{F}}^{\hat{d}}_n = \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{(n+1)*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}. \quad \square$$

Proof. By recurrence on n.

— For the basis $n = 0$, we have $\{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{1*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$

$= \varnothing = \hat{\mathsf{F}}^{\hat{d}}_0$ ≀def. $\overset{1*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} = \mathbb{1}_\Pi$, $(\vdash \xrightarrow{\boxed{A}} \dashv) \notin \Theta$ and def. iterates≀

— For the induction step, assuming Lemma 20 for $n \geqslant 0$, we have

$$\begin{aligned}
& \hat{\mathsf{F}}^{\hat{d}}_{n+1} = \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!](\hat{\mathsf{F}}^{\hat{d}}_n) && \wr\text{def. iterates}\wr \\
= & \bigcup_{A\to\sigma\in\mathcal{R}} \{\vdash \xrightarrow{(\!|A} \pi \xrightarrow{A|\!)} \dashv \mid \check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \bullet\sigma] \overset{(n+1)*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi \wedge \pi \in \Theta\} && \wr\text{def. (7) of } \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \text{ and Lemma 19}\wr \\
= & \bigcup_{A\to\sigma\in\mathcal{R}} \{\pi \in \Theta \mid \langle\vdash, \dashv\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A \to \sigma] \overset{(n+1)*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\} \\
& && \wr\text{def. (15) of } \boxed{D}\!\!\Longrightarrow_{\mathcal{G}}, \langle\vdash, \dashv\rangle \uparrow \bullet, \check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \bullet\sigma], \text{and (11) of } \check{\mathsf{R}}^{\check{D}}[A \to \sigma]\wr \\
= & \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{(n+2)*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\} && \wr\text{def. (15) of } \boxed{D}\!\!\Longrightarrow_{\mathcal{G}} \text{ and } \overset{(n+2)*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}\wr \\
= & \hat{\mathsf{F}}^{\hat{d}}_{n+1} && \wr\text{def. } \hat{\mathsf{F}}^{\hat{d}}_{n+1}\wr. \quad \square
\end{aligned}$$

Theorem 21. $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] = \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}$ $\square$

Proof.

$$\begin{aligned}
& \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\subseteq} \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] = \bigcup_{n\in\mathbb{N}} \hat{\mathsf{F}}^{\hat{d}}_n \\
& && \wr\text{by Theorem 14 where } \hat{\mathsf{F}}^{\hat{d}}_n, n \in \mathbb{N} \text{ are the iterates of } \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \text{ since } \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \text{ preserves lubs}\wr \\
= & \bigcup_{n\in\mathbb{N}} \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{(n+1)*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\} && \wr\text{by Lemma 20}\wr \\
= & \{\pi \in \Theta \mid \exists A \in \mathcal{N} : \bigvee_{n\geqslant 0} (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{n*}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\} && \wr\text{since } \vdash \xrightarrow{\boxed{A}} \dashv \neq \pi \in \Theta \text{ and def. } \textstyle\bigcup\wr \\
= & \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\} && \wr\text{def. } \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}\wr. \quad \square
\end{aligned}$$

12.2. Abstraction of the maximal protoderivation semantics into the maximal derivation semantics

Let us define the abstraction

$$\alpha^{\check{D}\hat{d}} \triangleq \boldsymbol{\lambda} P \bullet \boldsymbol{\lambda} A \bullet P(A) \cap \Theta \tag{17}$$

which collects the terminal traces (without nonterminal variables) among prototraces. This abstraction defines a Galois connection [27] $\langle \mathcal{N} \mapsto \wp(\Pi), \dot{\subseteq}\rangle \underset{\alpha^{\check{D}\hat{d}}}{\overset{\gamma^{\check{D}\hat{d}}}{\longleftarrow\!\!\!\!\longrightarrow\!\!\!\!\!\rightarrow}} \langle \mathcal{N} \mapsto \wp(\Theta), \dot{\subseteq}\rangle$. The restriction of the top-down maximal protoderivation semantics is the maximal derivation semantics.

Theorem 22. $\alpha^{\check{D}\hat{d}}(\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]) = \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].A$ $\square$

Proof.

$$\begin{aligned}
& \alpha^{\check{D}\hat{d}}(\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]) \\
= & \boldsymbol{\lambda} A \bullet \{\pi \in \Theta \mid (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\} && \wr\text{def. (16) of } \mathsf{S}^{\check{D}}[\![\mathcal{G}]\!], \text{def. } \alpha^{\check{D}\hat{d}}, \text{and } \Theta \subseteq \Pi\wr \\
= & \boldsymbol{\lambda} A \bullet \{\pi \in \Theta \mid \exists A \in \mathcal{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}} \pi\}.A \\
& && \wr\text{def. selection } \bullet.A \text{ and } \pi \text{ is a trace for } A \text{ by def. (15) of } \boxed{D}\!\!\Longrightarrow_{\mathcal{G}} \text{ and } \overset{\star}{\boxed{D}\!\!\Longrightarrow}_{\mathcal{G}}\wr \\
= & \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].A && \wr\text{Theorem 21}\wr. \quad \square
\end{aligned}$$

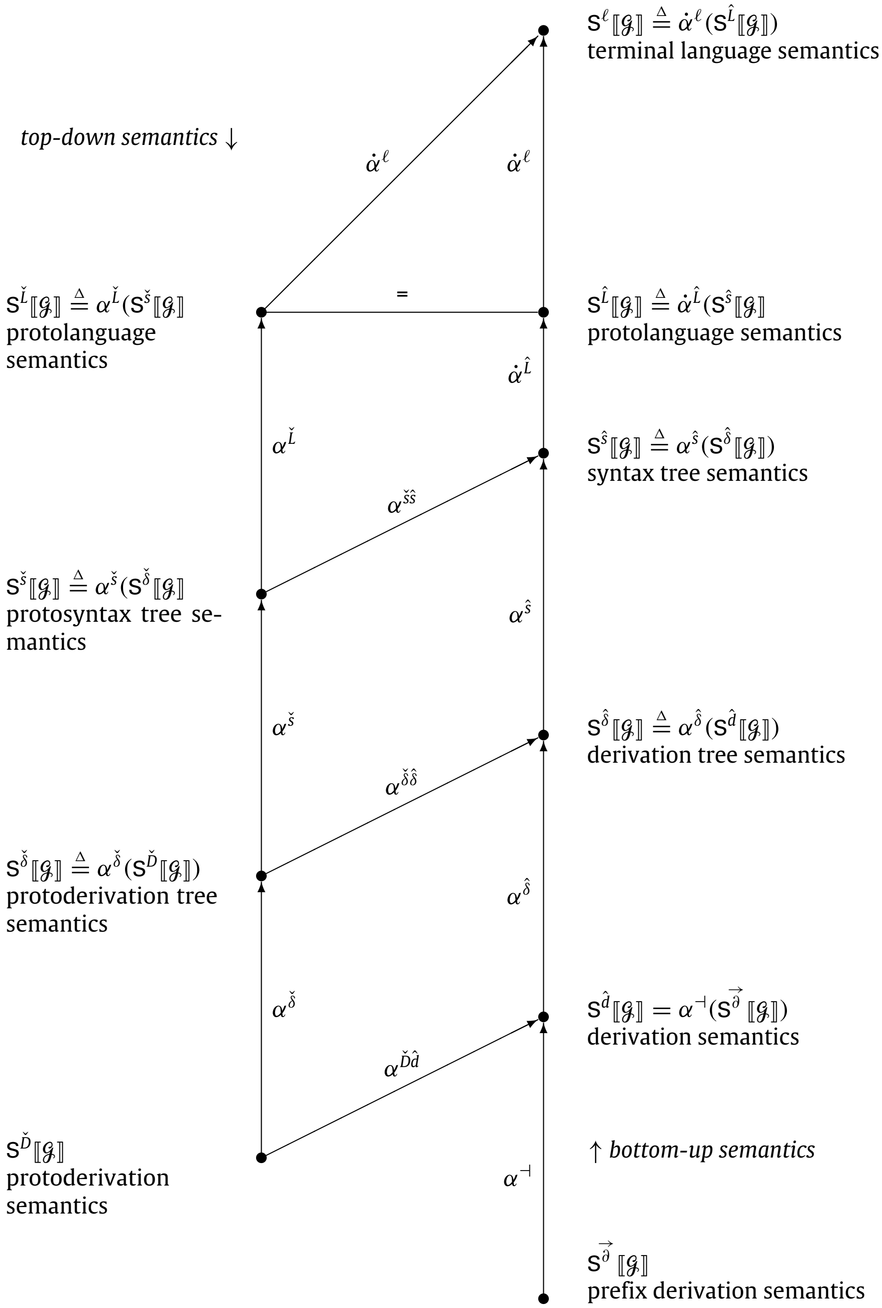


Fig. 1. The hierarchy of grammar semantics.

13. The hierarchy of grammar semantics

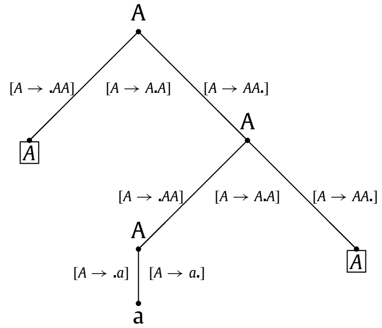
Theorem 22 shows that the bottom-up derivation semantics $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!]$ of a grammar $\mathcal{G}$ is, up to an isomorphism, an abstraction of the top-down protoderivation semantics $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!] \triangleq \boldsymbol{\lambda} A \bullet \{\pi \in \Pi \mid (\vdash \xrightarrow{\boxed{A}} \dashv) \overset{\star}{\underset{\mathcal{G}}{\boxed{D}\!\Longrightarrow}} \pi\}$ by the abstraction $\alpha^{\check{D}\hat{d}}$. We now introduce a hierarchy of abstractions of the protoderivation semantics $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]$, as given in Fig. 1. The various semantics and abstractions in Fig. 1, (apart from $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]$ (16), $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!]$ (5), and $\alpha^{\check{D}\hat{d}}$ (17) which have already been defined), are described below.

13.1. [Proto]derivation tree abstraction $\alpha^{\check{\delta}}$ and $\alpha^{\hat{\delta}}$

13.1.1. [Proto]derivation trees

[Proto]derivations can be described by [proto]derivation trees where internal nodes are labelled with nonterminals, leafs are labelled with terminals [or nonterminal variables] and branches are decorated with rule states.

Example 23. One possible protoderivation tree for the protosentence AaA of the grammar $\langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$ is given on the right. It can be represented in parenthesized form through an infix traversal as $(\!| A[A \to \bullet AA]\ \boxed{A}\ [A \to A\bullet A](\!| A[A \to \bullet AA](\!| A[A \to \bullet a]a[A \to a\bullet]A |\!)[A \to A\bullet A]\boxed{A}[A \to AA\bullet]A|\!)[A \to AA\bullet]A|\!)$ $\square$



We let $\check{\mathscr{U}} \triangleq \mathscr{T} \cup \mathscr{N}^{\square} \cup \mathscr{R}^{\cdot}$ and $\check{\mathfrak{D}} \triangleq (\mathscr{P} \cup \check{\mathscr{U}})^{\star}$. A *protoderivation tree* $\check{\delta}$ is represented by a well-parenthesized sentence over $\check{\mathscr{U}}$ so that $\check{\delta} \in \mathbb{P}_{\mathscr{P},\check{\mathscr{U}}} \subseteq \check{\mathfrak{D}}$. We extend the selection to $\wp(\check{\mathfrak{D}})$ whence $\wp(\mathbb{P}_{\mathscr{P},\check{\mathscr{U}}})$ as $D.A \triangleq \{(\!| B\sigma B |\!) \in D \mid B = A\} \cup \{\boxed{B} \in D \mid B = A\}$ so that $D.A$ is the set of protoderivation trees in D rooted at $A \in \mathscr{N}$.

13.1.2. Protoderivation tree abstraction $\alpha^{\check{\delta}}$ of protoderivations

The *protoderivation tree abstraction* $\alpha^{\check{\delta}} \in \Pi \mapsto \check{\mathfrak{D}}$ of protoderivations is

$$\begin{array}{rcl@{\qquad}rcl}
\alpha^{\check{\delta}}(\varpi \xrightarrow{\kappa} \tau) & \triangleq & \alpha^{\check{\delta}}(\varpi)\kappa\alpha^{\check{\delta}}(\tau) & \alpha^{\check{\delta}}(\dashv) & \triangleq & \epsilon \\
\alpha^{\check{\delta}}(\epsilon) & \triangleq & \epsilon & \alpha^{\check{\delta}}(s_1 \ldots s_n) & \triangleq & s_n, \quad s_1 \ldots s_n \in \mathscr{S}, \\
\alpha^{\check{\delta}}(\vdash) & \triangleq & \epsilon & & & n > 0, \quad \text{otherwise}
\end{array}$$

which is extended elementwise to $\alpha^{\check{\delta}} \in \wp(\Pi) \mapsto \wp(\check{\mathfrak{D}})$ as $\alpha^{\check{\delta}}(T) \triangleq \{\alpha^{\check{\delta}}(\pi) \mid \pi \in T\}$ so that we get the Galois connection $\langle \wp(\Pi),\ \subseteq\rangle \xrightleftharpoons[\alpha^{\check{\delta}}]{\gamma^{\check{\delta}}} \langle \wp(\check{\mathfrak{D}}),\ \subseteq\rangle$, further extended pointwise to $\alpha^{\check{\delta}} \in (\mathscr{N} \mapsto \wp(\Pi)) \mapsto (\mathscr{N} \mapsto \wp(\check{\mathfrak{D}}))$ as $\alpha^{\check{\delta}}(\phi) \triangleq \boldsymbol{\lambda} A \bullet \alpha^{\check{\delta}}(\phi(A))$.

13.1.3. Derivation tree abstraction $\alpha^{\hat{\delta}}$ of derivations

The restriction of $\alpha^{\check{\delta}}$ to derivation trees $\hat{\mathfrak{D}} \triangleq (\mathscr{P} \cup \hat{\mathscr{U}})^{\star}$ where $\hat{\mathscr{U}} \triangleq \mathscr{T} \cup \mathscr{R}^{\cdot}$ is $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathfrak{D}}$ such that

$$\begin{array}{rcl@{\qquad}rcl}
\alpha^{\hat{\delta}}(\epsilon) & \triangleq & \epsilon & \alpha^{\hat{\delta}}(\varpi \xrightarrow{\ell} \theta) & \triangleq & \alpha^{\hat{\delta}}(\varpi)\ell\alpha^{\hat{\delta}}(\theta) \\
\alpha^{\hat{\delta}}(\vdash) & \triangleq & \epsilon & & & \\
\alpha^{\hat{\delta}}(\dashv) & \triangleq & \epsilon & \alpha^{\hat{\delta}}(s_1 \ldots s_n) & \triangleq & s_n, \quad s_1 \ldots s_n \in \mathscr{S},\ n > 0, \quad \text{otherwise}
\end{array}$$

which is extended elementwise to $\alpha^{\hat{\delta}} \in \wp(\Theta) \mapsto \wp(\hat{\mathfrak{D}})$ as $\forall T \in \wp(\Theta) : \alpha^{\hat{\delta}}(T) \triangleq \{\alpha^{\hat{\delta}}(\theta) \mid \theta \in T\}$ so that we get a Galois connection between sets of traces and sets of derivation trees, as follows $\langle \wp(\Theta),\ \subseteq\rangle \xrightleftharpoons[\alpha^{\hat{\delta}}]{\gamma^{\hat{\delta}}} \langle \wp(\hat{\mathfrak{D}}),\ \subseteq\rangle$.

A *derivation tree* $\hat{\delta}$ is represented by a well-parenthesized sentence over $\hat{\mathscr{U}}$ so that $\hat{\delta} \in \mathbb{P}_{\mathscr{P},\hat{\mathscr{U}}} \subseteq \hat{\mathfrak{D}}$.

Lemma 24. *If T is a set of derivations then*
$\alpha^{\hat{\delta}}(\langle\varpi,\ \varpi'\rangle \uparrow T) = \{\alpha^{\hat{\delta}}(\varpi)\alpha^{\hat{\delta}}(\tau)\alpha^{\hat{\delta}}(\varpi') \mid \tau \in T\}$ $\square$

Proof. For a derivation $\vdash \xrightarrow{\ell_0} \dashv\varpi_1 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv$, we have

$$\begin{array}{cll}
& \alpha^{\hat{\delta}}(\langle\varpi,\ \varpi'\rangle \uparrow \vdash \xrightarrow{\ell_0} \dashv\varpi_1 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv) & \\
= & \alpha^{\hat{\delta}}(\varpi)\alpha^{\hat{\delta}}(\vdash)\ell_0\alpha^{\hat{\delta}}(\varpi_1) \ldots \alpha^{\hat{\delta}}(\varpi_{n-1})\ell_{n-1}\alpha^{\hat{\delta}}(\dashv)\alpha^{\hat{\delta}}(\varpi') & \wr \text{def. } \langle\varpi,\ \varpi'\rangle \uparrow \theta, \alpha^{\hat{\delta}}, \text{and } \alpha^{\hat{\delta}} \int \\
= & \alpha^{\hat{\delta}}(\varpi)\alpha^{\hat{\delta}}(\vdash \xrightarrow{\ell_0} \dashv\varpi_1 \ldots \dashv\varpi_{n-1} \xrightarrow{\ell_{n-1}} \dashv)\alpha^{\hat{\delta}}(\varpi') & \wr \text{def. } \alpha^{\hat{\delta}} \int
\end{array}$$

It follows that for a set T of derivations, we have $\alpha^{\hat{\delta}}(\langle\varpi,\ \varpi'\rangle \uparrow T)$

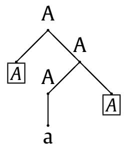
$$\begin{array}{cll}
= & \{\alpha^{\hat{\delta}}(\langle\varpi,\ \varpi'\rangle \uparrow \tau) \mid \tau \in T\} & \wr \text{def. } \langle\varpi,\ \varpi'\rangle \uparrow \theta \text{ and } \alpha^{\hat{\delta}} \int \\
= & \{\alpha^{\hat{\delta}}(\varpi)\alpha^{\hat{\delta}}(\tau)\alpha^{\hat{\delta}}(\varpi') \mid \tau \in T\} & \wr \text{as shown above} \int. \quad \square
\end{array}$$

13.2. [Proto]syntax tree abstraction $\alpha^{\check{s}}$ and $\alpha^{\hat{s}}$

13.2.1. Protosyntax trees

[Proto]syntax trees are [proto]-derivation trees denuded of the rule states decorating the branches. We represent [proto]syntax trees in parenthesized form through an infix traversal. We let $\check{\mathscr{T}} \triangleq (\mathscr{P} \cup \mathscr{T} \cup \mathscr{N}^{\square})^{\star}$. A *protosyntax tree* $\check{\tau}$ is represented by a well-parenthesized sentence over $(\mathscr{T} \cup \mathscr{N}^{\square})$ so that $\check{\tau} \in \mathbb{P}_{\mathscr{P},(\mathscr{T}\cup\mathscr{N}^{\square})} \subseteq \check{\mathscr{T}}$.

Example 25. One possible protosyntax tree for the protosentence AaA of the grammar $\langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$ is given on the right and represented as $\llparenthesis A\boxed{A}\llparenthesis A\llparenthesis AaA\rrparenthesis\boxed{A}A\rrparenthesis A\rrparenthesis$. $\square$



13.2.2. Protosyntax tree abstraction $\alpha^{\check{s}}$ of protoderivation trees

The *protosyntax tree abstraction* $\alpha^{\check{s}} \in \check{\mathscr{D}} \mapsto \check{\mathscr{T}}$ of protoderivation trees is ($A \in \mathscr{N}$, $\ell \in \mathscr{L}$)

$$\begin{array}{rcl@{\qquad}rcl}
\alpha^{\check{s}}(\sigma\llparenthesis A\sigma') & \triangleq & \alpha^{\check{s}}(\sigma)\llparenthesis A\alpha^{\check{s}}(\sigma') & \alpha^{\check{s}}(\sigma[A \to \varsigma\centerdot\varsigma']\sigma') & \triangleq & \alpha^{\check{s}}(\sigma)\alpha^{\check{s}}(\sigma') \\
\alpha^{\check{s}}(\sigma A\rrparenthesis\sigma') & \triangleq & \alpha^{\check{s}}(\sigma)A\rrparenthesis\alpha^{\check{s}}(\sigma') & \alpha^{\check{s}}(\sigma\ell\sigma') & \triangleq & \alpha^{\check{s}}(\sigma)\ell\alpha^{\check{s}}(\sigma') \\
\alpha^{\check{s}}(\sigma\boxed{A}\sigma') & \triangleq & \alpha^{\check{s}}(\sigma)\boxed{A}\alpha^{\check{s}}(\sigma') & \alpha^{\check{s}}(\epsilon) & \triangleq & \epsilon
\end{array}$$

extended elementwise to $\alpha^{\check{s}} \in \wp(\check{\mathscr{D}}) \mapsto \wp(\check{\mathscr{T}})$ as $\alpha^{\check{s}}(D) \triangleq \{\alpha^{\check{s}}(\check{\delta}) \mid \check{\delta} \in D\}$ so that we get a Galois connection $\langle\wp(\check{\mathscr{D}}),\ \subseteq\rangle \underset{\alpha^{\check{s}}}{\overset{\gamma^{\check{s}}}{\leftarrows\!\!\!\twoheadrightarrow}} \langle\wp(\check{\mathscr{T}}),\ \subseteq\rangle$ which can be extended pointwise to $(\mathscr{N} \mapsto \wp(\check{\mathscr{D}})) \mapsto (\mathscr{N} \mapsto \wp(\check{\mathscr{T}}))$ as $\alpha^{\check{s}}(\phi) \triangleq \boldsymbol{\lambda} A \bullet \alpha^{\check{s}}(\phi(A))$ so that $\langle\mathscr{N} \mapsto \wp(\check{\mathscr{D}}),\ \dot{\subseteq}\rangle \underset{\alpha^{\check{s}}}{\overset{\gamma^{\check{s}}}{\leftarrows\!\!\!\twoheadrightarrow}} \langle\mathscr{N} \mapsto \wp(\check{\mathscr{T}}),\ \dot{\subseteq}\rangle$.

13.2.3. Syntax tree abstraction $\alpha^{\hat{s}}$ of derivation trees

The restriction $\alpha^{\hat{s}}$ to syntax trees $\hat{\mathscr{T}} \triangleq (\mathscr{P} \cup \mathscr{T})^{\star}$ is $\alpha^{\hat{s}} \in \hat{\mathscr{D}} \mapsto \hat{\mathscr{T}}$ such that ($A \in \mathscr{N}$, $\ell \in \mathscr{L}$)

$$\begin{array}{rcl@{\qquad}rcl}
\alpha^{\hat{s}}(\sigma\llparenthesis A\sigma') & \triangleq & \alpha^{\hat{s}}(\sigma)\llparenthesis A\alpha^{\hat{s}}(\sigma') & \alpha^{\hat{s}}(\sigma[A \to \varsigma\centerdot\varsigma']\sigma') & \triangleq & \alpha^{\hat{s}}(\sigma)\alpha^{\hat{s}}(\sigma') \\
\alpha^{\hat{s}}(\sigma A\rrparenthesis\sigma') & \triangleq & \alpha^{\hat{s}}(\sigma)A\rrparenthesis\alpha^{\hat{s}}(\sigma') & \alpha^{\hat{s}}(\sigma\ell\sigma') & \triangleq & \alpha^{\hat{s}}(\sigma)\ell\alpha^{\hat{s}}(\sigma') \\
\alpha^{\hat{s}}(\epsilon) & \triangleq & \epsilon & & &
\end{array}$$

extended elementwise to $\alpha^{\hat{s}} \in \wp(\hat{\mathscr{D}}) \mapsto \wp(\hat{\mathscr{T}})$ as $\alpha^{\hat{s}}(D) \triangleq \{\alpha^{\hat{s}}(\hat{\delta}) \mid \hat{\delta} \in D\}$ so that we get a Galois connection between sets of derivation trees and sets of syntax trees, as follows $\langle\wp(\hat{\mathscr{D}}),\ \subseteq\rangle \underset{\alpha^{\hat{s}}}{\overset{\gamma^{\hat{s}}}{\leftarrows\!\!\!\twoheadrightarrow}} \langle\wp(\hat{\mathscr{T}}),\ \subseteq\rangle$. A *syntax tree* $\hat{\tau}$ is represented by a well-parenthesized sentence over $\mathscr{T}$ so that $\hat{\tau} \in \mathbb{P}_{\mathscr{P},\mathscr{T}} \subseteq \hat{\mathscr{T}}$.

13.3. Protosentence abstraction $\alpha^{\check{L}}$ and $\dot{\alpha}^{\hat{L}}$

13.3.1. Protolanguages

The *protolanguage* of a grammar $\mathcal{G} = \langle\mathscr{T},\ \mathscr{N},\ \overline{S},\ \mathscr{R}\rangle$ with $\mathscr{V} \triangleq \mathscr{T} \cup \mathscr{N}$ is the set of protosentences deriving from the grammar axiom $\overline{S}$ where *protosentences* $\eta \in \mathscr{V}^{\star}$ contain both terminals in $\mathscr{T}$ and nonterminals in $\mathscr{N}$ and the derivation consists in replacing a nonterminal A by the right-hand side σ of a grammar rule $A \to \sigma \in \mathscr{R}$.

13.3.2. Protosentence abstraction $\alpha^{\check{L}}$ of protosyntax trees

The *protolanguage abstraction* $\alpha^{\check{L}} \in \check{\mathscr{T}} \mapsto \mathscr{V}^{\star}$ of protosyntax trees is defined as (we follow the tradition of confusing nonterminals A denoting the grammatical structure and nonterminal variables $\boxed{A}$ for protosentence substitution since confusion between attributes of internal tree nodes in $\mathscr{N}$ and variables in $\mathscr{N}^{\square}$ is no longer possible)

$$\begin{array}{rcl@{\qquad}rcl}
\alpha^{\check{L}}(\sigma\llparenthesis A\sigma') & \triangleq & \alpha^{\check{L}}(\sigma)\alpha^{\check{L}}(\sigma'), \quad A \in \mathscr{N} & \alpha^{\check{L}}(\sigma a\sigma') & \triangleq & \alpha^{\check{L}}(\sigma)a\alpha^{\check{L}}(\sigma'), \quad a \in \mathscr{T} \\
\alpha^{\check{L}}(\sigma A\rrparenthesis\sigma') & \triangleq & \alpha^{\check{L}}(\sigma)\alpha^{\check{L}}(\sigma') & \alpha^{\check{L}}(\epsilon) & \triangleq & \epsilon \\
\alpha^{\check{L}}(\sigma\boxed{A}\sigma') & \triangleq & \alpha^{\check{L}}(\sigma)A\alpha^{\check{L}}(\sigma') & & &
\end{array}$$

extended elementwise to $\alpha^{\check{L}} \in \wp(\check{\mathcal{T}}) \mapsto \wp(\mathcal{V}^\star)$ as $\alpha^{\check{L}}(D) \triangleq \{\alpha^{\check{L}}(\check{\tau}) \mid \check{\tau} \in D\}$ so that we get a Galois connection $\langle \wp(\check{\mathcal{T}}), \subseteq\rangle \xrightleftharpoons[\alpha^{\check{L}}]{\gamma^{\check{L}}} \langle \wp(\mathcal{V}^\star), \subseteq\rangle$ which can be extended pointwise to $\alpha^{\check{L}} \in (\mathcal{N} \mapsto \wp(\check{\mathcal{T}})) \mapsto (\mathcal{N} \mapsto \wp(\mathcal{V}^\star))$ as $\alpha^{\check{L}}(\phi) \triangleq \boldsymbol{\lambda} A \bullet \alpha^{\check{L}}(\phi(A))$ such that $\langle \mathcal{N} \mapsto \wp(\check{\mathcal{T}}),\ \dot{\subseteq}\rangle \xrightleftharpoons[\alpha^{\check{L}}]{\gamma^{\check{L}}} \langle \mathcal{N} \mapsto \wp(\mathcal{V}^\star),\ \dot{\subseteq}\rangle$.

Example 26. For the protosyntax tree in Example 25 of the grammar $\langle \{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$, we have $\alpha^{\check{L}}\big((\!|A\boxed{A}(\!|A(\!|AaA|\!)\boxed{A}A|\!)A|\!)\big) = AaA$. $\square$

13.3.3. Protosentence abstraction $\dot{\alpha}^{\hat{L}}$ of syntax trees

For syntax trees, we define the flattener $\alpha^{\hat{L}} \in \hat{\mathcal{T}} \mapsto \wp(\mathcal{V}^\star)$ as

$$\alpha^{\hat{L}}((\!|A\sigma A|\!)\sigma') \triangleq (\{A\} \cup \alpha^{\hat{L}}(\sigma))\alpha^{\hat{L}}(\sigma') \quad \alpha^{\hat{L}}(a\sigma') \triangleq \{a\}\alpha^{\hat{L}}(\sigma') \quad \alpha^{\hat{L}}(\epsilon) \triangleq \{\epsilon\}$$

extended elementwise to $\alpha^{\hat{L}} \in \wp(\hat{\mathcal{T}}) \mapsto \wp(\mathcal{V}^\star)$ as $\alpha^{\hat{L}}(\Sigma) \triangleq \bigcup\{\alpha^{\hat{L}}(\sigma) \mid \sigma \in \Sigma\}$ and pointwise to $\dot{\alpha}^{\hat{L}} \in \wp(\hat{\mathcal{T}}) \mapsto (\mathcal{N} \mapsto \wp(\mathcal{V}^\star))$ as $\dot{\alpha}^{\hat{L}}(S) \triangleq \boldsymbol{\lambda} A \bullet \alpha^{\hat{L}}(S.A)$ so that we get the Galois connection $\langle \wp(\hat{\mathcal{T}}),\ \subseteq\rangle \xrightleftharpoons[\dot{\alpha}^{\hat{L}}]{\dot{\gamma}^{\hat{L}}} \langle \mathcal{N} \mapsto \wp(\mathcal{V}^\star),\ \dot{\subseteq}\rangle$.

13.4. Terminal sentence abstraction $\dot{\alpha}^{\ell}$

13.4.1. Languages

The classical semantics of a context-free grammar $\mathcal{G} = \langle \mathcal{T},\ \mathcal{N},\ \overline{S},\ \mathcal{R}\rangle$ is a set of finite terminal sentences in $\wp(\mathcal{T}^\star)$ [10, 11].

13.4.2. Terminal sentence abstraction $\dot{\alpha}^{\ell}$ of protolanguages

Terminal sentence abstraction eliminates the sentences of a protolanguage which are not terminal. Let us define the eraser $\alpha^{\ell} \in \mathcal{V}^\star \mapsto \wp(\mathcal{T}^\star)$ as

$$\alpha^{\ell}(A\sigma) \triangleq \varnothing \qquad \alpha^{\ell}(a\sigma) \triangleq a\alpha^{\ell}(\sigma) \qquad \alpha^{\ell}(\epsilon) \triangleq \epsilon$$

extended to $\alpha^{\ell} \in \wp(\mathcal{V}^\star) \mapsto \wp(\mathcal{T}^\star)$ as $\alpha^{\ell}(\Sigma) \triangleq \bigcup\{\alpha^{\ell}(\sigma) \mid \sigma \in \Sigma\} = \Sigma \cap \mathcal{T}^\star$ so that we get a Galois connection $\langle \wp(\mathcal{V}^\star),\ \subseteq\rangle \xrightleftharpoons[\alpha^{\ell}]{\gamma^{\ell}} \langle \wp(\mathcal{T}^\star),\ \subseteq\rangle$ which can be extended pointwise to $\dot{\alpha}^{\ell} \in (\mathcal{N} \mapsto \wp(\mathcal{V}^\star)) \mapsto (\mathcal{N} \mapsto \wp(\mathcal{T}^\star))$ as $\dot{\alpha}^{\ell}(\rho) \triangleq \boldsymbol{\lambda} A \bullet \alpha^{\ell}(\rho(A))$ such that $\langle \mathcal{N} \mapsto \wp(\mathcal{V}^\star),\ \dot{\subseteq}\rangle \xrightleftharpoons[\dot{\alpha}^{\ell}]{\dot{\gamma}^{\ell}} \langle \mathcal{N} \mapsto \wp(\mathcal{T}^\star),\ \dot{\subseteq}\rangle$.

14. Fixpoint bottom-up structural abstract semantics

14.1. Bottom-up abstract interpreter

All bottom-up semantics $\mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!] \in \hat{\mathsf{D}}^{\hat{\sharp}}$ of context-free grammars $\mathcal{G}$ are instances of the following abstract interpreter (which generalizes the bottom-up grammar flow analysis of [8, Def. 8.2.18]).

$$\mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\sqsubseteq}\, \hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!] \tag{18}$$

where $\langle \hat{\mathsf{D}}^{\hat{\sharp}},\ \sqsubseteq,\ \bot,\ \sqcup\rangle$ is a cpo/complete lattice and the transformer $\hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!] \in \hat{\mathsf{D}}^{\hat{\sharp}} \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}$ is

$$\hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!] \triangleq \boldsymbol{\lambda} \rho \bullet \bigsqcup_{A \to \sigma \in \mathcal{R}} A^{\hat{\sharp}}(\hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot}[A \to \boldsymbol{.}\sigma]\rho) \tag{19}$$

while $\langle \hat{\mathsf{D}}^{\hat{\sharp}}_{\cdot},\ \sqsubseteq_{\cdot},\ \bot_{\cdot},\ \sqcup_{\cdot}\rangle$ is a cpo/complete lattice, and the transformer $\hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot} \in \mathcal{R} \mapsto \hat{\mathsf{D}}^{\hat{\sharp}} \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}_{\cdot}$ is

$$\hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot}[A \to \sigma\boldsymbol{.}a\sigma'] \triangleq \boldsymbol{\lambda} \rho \bullet [A \to \sigma\boldsymbol{.}a\sigma']^{\hat{\sharp}} \,\mathring{,}^{\hat{\sharp}}\, \hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot}[A \to \sigma a\boldsymbol{.}\sigma']\rho \tag{20}$$

$$\hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot}[A \to \sigma\boldsymbol{.}B\sigma'] \triangleq \boldsymbol{\lambda} \rho \bullet [A \to \sigma\boldsymbol{.}B\sigma']^{\hat{\sharp}}(\rho, B) \,\mathring{\varsigma}^{\hat{\sharp}}\, \hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot}[A \to \sigma B\boldsymbol{.}\sigma']\rho \tag{21}$$

$$\hat{\mathsf{F}}^{\hat{\sharp}}_{\cdot}[A \to \sigma\boldsymbol{.}] \triangleq \boldsymbol{\lambda} \rho \bullet [A \to \sigma\boldsymbol{.}]^{\hat{\sharp}} \tag{22}$$

with	$A^{\hat{\sharp}} \in \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}} \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}$	abstract rooting
	$[A \to \sigma\,\boldsymbol{.}a\sigma']^{\hat{\sharp}} \in \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}$	terminal abstraction
	$\mathring{\,}^{\hat{\sharp}} \in (\hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}} \times \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}) \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}$	abstract concatenation
	$[A \to \sigma\,\boldsymbol{.}B\sigma']^{\hat{\sharp}} \in (\hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}} \times \mathscr{N}) \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}$	nonterminal abstraction
	$\fatsemi^{\hat{\sharp}} \in (\hat{\mathsf{D}}^{\hat{\sharp}} \times \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}) \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}$	abstract junction
	$[A \to \sigma\,\boldsymbol{.}]^{\hat{\sharp}} \in \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}$	emptiness abstraction.

Observe that Theorem 14 is an instance of (18) where $\hat{\mathsf{D}}^{\hat{\sharp}} = \hat{\mathsf{D}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}$ is $\wp(\Theta)$, $\hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!]$ (19) is the set of traces bottom-up transformer $\hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!] \in \wp(\Theta) \mapsto \wp(\Theta)$ (7), and $\hat{\mathsf{F}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}[A \to \sigma\,\boldsymbol{.}\sigma']$ is $\hat{\mathsf{F}}^{\hat{d}}_{\boldsymbol{\cdot}}[A \to \sigma\,\boldsymbol{.}\sigma'] \in \wp(\Theta) \mapsto \wp(\Theta)$ as defined in (8)–(10), which is exactly of the form (20)–(22).

14.2. Well-definedness of the bottom-up abstract interpreter

The existence of the least fixpoint is guaranteed by the following

Hypothesis 27. For all $[A \to \sigma\,\boldsymbol{.}\sigma'] \in \mathscr{R}^{\boldsymbol{\cdot}}$, $\boldsymbol{\lambda}\,\rho \bullet A^{\hat{\sharp}}(\hat{\mathsf{F}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}[A \to \boldsymbol{.}\sigma]\rho) \in \hat{\mathsf{D}}^{\hat{\sharp}} \mapsto \hat{\mathsf{D}}^{\hat{\sharp}}$ is upper continuous for the ordering $\sqsubseteq$ on $\hat{\mathsf{D}}^{\hat{\sharp}}$.[4] □

Hypothesis 27 is guaranteed by the following *local continuity conditions*.

Lemma 28. *If $A^{\hat{\sharp}}$ is continuous, $\mathring{\,}^{\hat{\sharp}}$ is continuous in its second argument, $[A \to \sigma\,\boldsymbol{.}B\sigma']^{\hat{\sharp}}$ is continuous in its first argument, $\fatsemi^{\hat{\sharp}}$ is continuous then Hypothesis 27 holds.* □

Proof sketch. The upper continuity of $\hat{\mathsf{F}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}[A \to \sigma\,\boldsymbol{.}\sigma']$, by induction on the length $|\sigma'|$ of σ'. □

14.3. Instances of the bottom-up abstract interpreter

The hierarchy of semantics discussed in Section 13 is obtained by the instances of the bottom-up abstract semantics (18) given in Fig. 2. Classical semantics and flow analyses also have the same form given in Fig. 3. These facts are proved in the following Section 15 for the bottom-up semantics and in Section 19 for bottom-up grammar flow analysis.

14.4. Soundness and completeness of the bottom-up abstract interpreter

Definition 29. An abstract semantics $\mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!] \in \hat{\mathsf{D}}^{\hat{\sharp}}$ is sound and complete with respect to a concrete semantics $\mathsf{S}^{\hat{\natural}}[\![\mathcal{G}]\!] \in \hat{\mathsf{D}}^{\hat{\sharp}}$ for an abstraction $\langle \hat{\mathsf{D}}^{\hat{\natural}},\ \sqsubseteq^{\hat{\natural}} \rangle \xleftrightarrow[\alpha]{\gamma} \langle \hat{\mathsf{D}}^{\hat{\sharp}},\ \sqsubseteq^{\hat{\sharp}} \rangle$. if and only if $\alpha(\mathsf{S}^{\hat{\natural}}[\![\mathcal{G}]\!]) = \mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!]$. □

This global soundness and completeness condition on the abstraction is implied by the *rule soundness and completeness condition*

$$\alpha(A^{\hat{\natural}}(\hat{\mathsf{F}}^{\hat{\natural}}_{\boldsymbol{\cdot}}[A \to \boldsymbol{.}\sigma]\rho)) \quad = \quad A^{\hat{\sharp}}(\hat{\mathsf{F}}^{\hat{\sharp}}_{\boldsymbol{\cdot}}[A \to \boldsymbol{.}\sigma]\alpha(\rho)). \tag{23}$$

Theorem 30. *The local soundness and completeness condition (23) implies the soundness and completeness of the abstract interpreter $\alpha(\mathsf{S}^{\hat{\natural}}[\![\mathcal{G}]\!]) = \alpha(\boldsymbol{lfp}^{\sqsubseteq^{\hat{\natural}}}\ \hat{\mathsf{F}}^{\hat{\natural}}[\![\mathcal{G}]\!]) = \boldsymbol{lfp}^{\sqsubseteq^{\hat{\sharp}}}\ \hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!] = \mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!]$.* □

Note 31. The local soundness and completeness condition (23) can be weakened according to the hypotheses of one of the fixpoint abstraction theorems of Appendix A.2 such as Corollary 101 or Corollary 106. □

Proof of Theorem 30. The main point is to show the commutation property

[4] Indeed monotony is sufficient [28].

Abstract semantics $\mathsf{S}^{\sharp}[\![\mathcal{G}]\!]$	Maximal derivation $\mathsf{S}^{d}[\![\mathcal{G}]\!]$	Derivation tree $\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!]$	Syntax tree $\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!]$	Proto-language $\mathsf{S}^{L}[\![\mathcal{G}]\!]$
$\hat{\mathsf{D}}^{\hat{\sharp}}$	$\wp(\Theta)$	$\wp(\hat{\mathcal{D}})$	$\wp(\hat{\mathcal{T}})$	$\mathcal{N} \mapsto \wp(\mathcal{V}^{\star})$
$\sqsubseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\dot{\subseteq}$
$\perp$	$\varnothing$	$\varnothing$	$\varnothing$	$\dot{\varnothing}$
$\sqcup$	$\cup$	$\cup$	$\cup$	$\dot{\cup}$
$\hat{\mathsf{D}}^{\hat{\sharp}}_{\bullet}$	$\wp(\Theta)$	$\wp(\hat{\mathcal{D}})$	$\wp(\hat{\mathcal{T}})$	$\wp(\mathcal{V}^{\star})$
$\sqsubseteq_{\bullet}$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$
$\perp_{\bullet}$	$\varnothing$	$\varnothing$	$\varnothing$	$\varnothing$
$\sqcup_{\bullet}$	$\cup$	$\cup$	$\cup$	$\cup$
$A^{\hat{\sharp}}(X)$	$\vdash \xrightarrow{(\!\lvert A} X \xrightarrow{A\rvert\!)} \dashv$	$(\!\lvert AXA\rvert\!)$	$(\!\lvert AXA\rvert\!)$	$A^{\hat{L}}(X)^{(1)}$
$[A \to \sigma\,\bullet a\sigma']^{\hat{\sharp}}$	$(\dashv[A \to \sigma\,\bullet a\sigma']) \xrightarrow{a}$	$[A \to \sigma\,\bullet a\sigma']a$	$a^{(2)}$	a
$\circ^{\hat{\sharp}}$	$.^{(3)}$	$.$	$.$	$.$
$[A \to \sigma\,\bullet B\sigma']^{\hat{\sharp}}(\rho, B)$	$[A \to \sigma\,\bullet B\sigma']^{\hat{d}}(\rho, B)^{(4)}$	$[A \to \sigma\,\bullet B\sigma']\ \rho.B$	$\rho.B$	$\{B\} \cup \rho(B)$
$\fatsemi^{\hat{\sharp}}$	$\fatsemi$	$.$	$.$	$.$
$[A \to \sigma\bullet]^{\hat{\sharp}}$	$\dashv[A \to \sigma\bullet]$	$[A \to \sigma\bullet]$	$\epsilon^{(2)}$	ϵ

where $(\!\lvert \mathtt{tt} \mathrel{?} a \mathrel{:} b \rvert\!) = a$, $(\!\lvert \mathtt{ff} \mathrel{?} a \mathrel{:} b \rvert\!) = b$, $(\!\lvert \mathtt{ff} \mathrel{?} a \parallel \mathtt{tt} \mathrel{?} b \mathrel{:} c \rvert\!) = b$, $(\!\lvert \mathtt{ff} \mathrel{?} a \parallel \mathtt{ff} \mathrel{?} b \mathrel{:} c \rvert\!) = c$, etc., $^{(1)}$ $A^{\hat{L}}(X) \triangleq \boldsymbol{\lambda} A' \bullet (\!\lvert A' = A \mathrel{?} \{A\} \cup X \mathrel{:} \varnothing \rvert\!)$, $^{(2)}$ a (and ϵ) is a shorthand for $\{a\}$ (and $\{\epsilon\}$), $^{(3)}$ sentence and language concatenation . is denoted by juxtaposition, extended pointwise, and $^{(4)}$ $[A \to \sigma\,\bullet B\sigma']^{\hat{d}}(\rho, B) \triangleq \langle \dashv[A \to \sigma\,\bullet B\sigma'],\ \dashv[A \to \sigma B\bullet\sigma'] \rangle \uparrow \rho.B$.

Fig. 2. Semantic instances of the abstract bottom-up grammar semantics (18).

Abstract semantics $\mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!]$	Terminal language $\mathsf{S}^{\ell}[\![\mathcal{G}]\!]$	First $\mathsf{S}^{1}[\![\mathcal{G}]\!]$	ϵ-Productivity $\mathsf{S}^{\epsilon}[\![\mathcal{G}]\!]$	Nonterminal productivity $\mathsf{S}^{\otimes}[\![\mathcal{G}]\!]$
$\hat{\mathsf{D}}^{\hat{\sharp}}$	$\mathcal{N} \mapsto \wp(\mathcal{T}^{\star})$	$\mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\epsilon\})$	$\mathcal{N} \mapsto \mathbb{B}^{(1)}$	$\mathcal{N} \mapsto \mathbb{B}$
$\sqsubseteq$	$\dot{\subseteq}$	$\dot{\subseteq}$	$\dot{\Longrightarrow}$	$\dot{\Longrightarrow}$
$\perp$	$\dot{\varnothing}$	$\dot{\varnothing}$	$\boldsymbol{\lambda} N \bullet \mathtt{ff}$	$\boldsymbol{\lambda} N \bullet \mathtt{ff}$
$\sqcup$	$\dot{\cup}$	$\dot{\cup}$	$\dot{\vee}$	$\dot{\vee}$
$\hat{\mathsf{D}}^{\hat{\sharp}}_{\bullet}$	$\wp(\mathcal{T}^{\star})$	$\wp(\mathcal{T} \cup \{\epsilon\})$	$\mathbb{B}$	$\mathbb{B}$
$\sqsubseteq_{\bullet}$	$\subseteq$	$\subseteq$	$\Longrightarrow$	$\Longrightarrow$
$\perp_{\bullet}$	$\varnothing$	$\varnothing$	$\mathtt{ff}$	$\mathtt{ff}$
$\sqcup_{\bullet}$	$\cup$	$\cup$	$\vee$	$\vee$
$A^{\hat{\sharp}}(X)$	$A^{\ell}(X)^{(2)}$	$A^{1}(X)^{(2)}$	$A^{\epsilon}(X)^{(3)}$	$A^{\otimes}(X)^{(3)}$
$[A \to \sigma\bullet a\sigma']^{\hat{\sharp}}$	a	a	$\mathtt{ff}$	$\mathtt{tt}$
$\circ^{\hat{\sharp}}$	$.$	$\dot{\oplus}^{1\,(4)}$	$\dot{\wedge}$	$\dot{\wedge}$
$[A \to \sigma\bullet B\sigma']^{\hat{\sharp}}(\rho, B)$	$\rho(B)$	$\rho(B)$	$\rho(B)$	$\rho(B)$
$\fatsemi^{\hat{\sharp}}$	$.$	$\dot{\oplus}^{1\,(4)}$	$\dot{\wedge}$	$\dot{\wedge}$
$[A \to \sigma\bullet]^{\hat{\sharp}}$	ϵ	ϵ	$\mathtt{tt}$	$\mathtt{tt}$

where $^{(1)}$ $\mathbb{B} \triangleq \{\mathtt{ff}, \mathtt{tt}\}$, $^{(2)}$ $A^{\ell}(X) = A^{1}(X) \triangleq \boldsymbol{\lambda} A' \bullet (\!\lvert A' = A \mathrel{?} X \mathrel{:} \varnothing \rvert\!)$, $^{(3)}$ $A^{\epsilon}(X) = A^{\otimes}(X) \triangleq \boldsymbol{\lambda} A' \bullet (\!\lvert A' = A \mathrel{?} X \mathrel{:} \mathtt{ff} \rvert\!)$, the first abstraction $\oplus^{1}$ of language concatenation is defined in **Lem. 72**, and $^{(4)}$ $\dot{\oplus}^{1}$ is its pointwise extension.

Fig. 3. Flow analysis instances of the abstract bottom-up grammar semantics (18).

$$\alpha(\hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!](\rho)) \quad = \quad \alpha\Big(\bigsqcup_{A \to \sigma \in \mathcal{R}}^{\hat{\sharp}} A^{\hat{\sharp}}(\hat{\mathsf{F}}^{\hat{\sharp}}_{\bullet}[A \to \bullet\sigma]\rho)\Big) \qquad \text{(def. (19) of } \hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!]\text{)}$$

$$= \bigsqcup_{A \to \sigma \in \mathcal{R}}^{\sharp} \alpha(A^{\hat{\sharp}}(\hat{\mathsf{F}}^{\hat{\sharp}}_{\bullet}[A \to \bullet\sigma]\rho)) \qquad \text{(}\alpha \text{ preserves lubs in Galois connections)}$$

$$= \bigsqcup_{A\to\sigma\in\mathscr{R}}^{\hat{\sharp}} A^{\hat{\sharp}}(\hat{\mathsf{F}}^{\hat{\sharp}}_{.}[A \to \boldsymbol{.}\sigma]\alpha(\rho)) \qquad \text{≀by local soundness cond. (24)≀}$$
$$= \hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!](\alpha(\rho)) \qquad \text{≀def. (19) of } \hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!]\text{≀}. \quad \square$$

The local soundness and completeness condition (23) is implied by the stronger *local soundness and completeness conditions* on the abstract operators, where $\langle \hat{\mathsf{D}}^{\hat{\natural}}_{.},\ \sqsubseteq^{\hat{\natural}}_{.}\rangle \underset{\alpha_{.}}{\overset{\gamma_{.}}{\longleftrightarrow}} \langle \hat{\mathsf{D}}^{\hat{\sharp}}_{.},\ \sqsubseteq^{\hat{\sharp}}_{.}\rangle$ and for all $\rho \in \hat{\mathsf{D}}^{\hat{\natural}}$ and $x, y \in \hat{\mathsf{D}}^{\hat{\natural}}_{.}$,

$$\begin{array}{rclrcl} \alpha(A^{\hat{\natural}}(x)) &=& A^{\hat{\sharp}}(\alpha_{.}(x)), & \alpha_{.}([A \to \sigma\boldsymbol{.}B\sigma']^{\hat{\natural}}(\rho, B)) &=& [A \to \sigma\boldsymbol{.}B\sigma']^{\hat{\sharp}}(\alpha(\rho), B), \\ \alpha_{.}([A \to \sigma\boldsymbol{.}a\sigma']^{\hat{\natural}}) &=& [A \to \sigma\boldsymbol{.}a\sigma']^{\hat{\sharp}}, & \alpha_{.}(x \mathbin{\hat{\natural}{\scriptstyle ;}} y) &=& \alpha_{.}(x) \mathbin{\hat{\sharp}{\scriptstyle ;}} \alpha_{.}(y), \\ \alpha_{.}(x \mathbin{\circ^{\hat{\natural}}} y) &=& \alpha_{.}(x) \mathbin{\circ^{\hat{\sharp}}} \alpha_{.}(y), & \alpha_{.}([A \to \sigma\boldsymbol{.}]^{\hat{\natural}}) &=& [A \to \sigma\boldsymbol{.}]^{\hat{\sharp}}. \end{array} \tag{24}$$

Corollary 32. *The above local soundness and completeness conditions* (24) *imply the soundness and completeness of the abstract interpreter* $\alpha(\mathsf{S}^{\hat{\natural}}[\![\mathcal{G}]\!]) = \alpha(\boldsymbol{lfp}^{\sqsubseteq^{\hat{\natural}}}\ \hat{\mathsf{F}}^{\hat{\natural}}[\![\mathcal{G}]\!]) = \boldsymbol{lfp}^{\sqsubseteq^{\hat{\sharp}}}\ \hat{\mathsf{F}}^{\hat{\sharp}}[\![\mathcal{G}]\!] = \mathsf{S}^{\hat{\sharp}}[\![\mathcal{G}]\!]$. $\square$

Proof sketch. We observe that

$$\alpha_{.}(\hat{\mathsf{F}}^{\hat{\natural}}_{.}[A \to \sigma\boldsymbol{.}\sigma']\rho) \quad = \quad \hat{\mathsf{F}}^{\hat{\sharp}}_{.}[A \to \sigma\boldsymbol{.}\sigma'](\alpha(\rho)) \tag{25}$$

and so Corollary 32 follows from Theorem 30 and (24). $\square$

We now consider the instances of the abstract bottom-up semantics given in Fig. 2. The grammar flow analysis instances in Fig. 3 are considered in Section 19.

15. The hierarchy of bottom-up grammar semantics

15.1. Fixpoint bottom-up derivation tree semantics

15.1.1. Derivation tree semantics

The *derivation tree semantics* $\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!] \in \wp(\hat{\mathcal{D}})$ of a context-free grammar $\mathcal{G} = \langle \mathscr{T},\ \mathscr{N},\ \bar{S},\ \mathscr{R}\rangle$, is the set of derivation trees generated by the grammar $\mathcal{G}$. It is defined as the derivation tree abstraction of the derivation semantics, as follows

$$\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!] \quad \triangleq \quad \alpha^{\hat{\delta}}(\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!]). \tag{26}$$

Lemma 33. $\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!] \in \mathbb{P}_{\mathscr{P},\hat{\mathscr{U}}}$. $\square$

Proof. By Lemma 5 and definition of $\alpha^{\hat{\delta}}$. $\square$

15.1.2. Fixpoint bottom-up structural derivation tree semantics

Let the transformer $\hat{\mathsf{F}}^{\hat{\delta}}[\![\mathcal{G}]\!] \in \wp(\hat{\mathcal{D}}) \mapsto \wp(\hat{\mathcal{D}})$ be defined as follows

$$\hat{\mathsf{F}}^{\hat{\delta}}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda} D \bullet \bigcup_{A\to\sigma\in\mathscr{R}} (\!|A\ \hat{\mathsf{F}}^{\hat{\delta}}_{.}[A \to \boldsymbol{.}\sigma]D\ A|\!) \tag{27}$$

where $\hat{\mathsf{F}}^{\hat{\delta}}_{.}[\![\mathcal{G}]\!] \in \mathscr{R} \mapsto \wp(\hat{\mathcal{D}}) \mapsto \wp(\hat{\mathcal{D}})$ is

$$\begin{array}{rcl} \hat{\mathsf{F}}^{\hat{\delta}}_{.}[A \to \sigma\boldsymbol{.}a\sigma'] &\triangleq& \boldsymbol{\lambda} D \bullet [A \to \sigma\boldsymbol{.}a\sigma']\ a\ \hat{\mathsf{F}}^{\hat{\delta}}_{.}[A \to \sigma a\boldsymbol{.}\sigma']D \\ \hat{\mathsf{F}}^{\hat{\delta}}_{.}[A \to \sigma\boldsymbol{.}B\sigma'] &\triangleq& \boldsymbol{\lambda} D \bullet [A \to \sigma\boldsymbol{.}B\sigma']\ D.B\ \hat{\mathsf{F}}^{\hat{\delta}}_{.}[A \to \sigma B\boldsymbol{.}\sigma']D \\ \hat{\mathsf{F}}^{\hat{\delta}}_{.}[A \to \sigma\boldsymbol{.}] &\triangleq& \boldsymbol{\lambda} D \bullet [A \to \sigma\boldsymbol{.}]. \end{array}$$

The derivation tree semantics of a grammar $\mathcal{G}$ can now be expressed in fixpoint form for transformer $\hat{\mathsf{F}}^{\hat{\delta}}[\![\mathcal{G}]\!]$ as follows.

Theorem 34.

$$\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!] \quad = \quad \boldsymbol{lfp}^{\subseteq}\ \hat{\mathsf{F}}^{\hat{\delta}}[\![\mathcal{G}]\!]. \quad \square$$

Example 35. The derivation tree semantics of the grammar $\langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$, is the least fixpoint of the equation

$$D \quad = \quad \begin{array}{l} \{(\!|A\ [A \to \textbf{.}a]\ a\ [A \to a\textbf{.}]\ A|\!)\} \\ \cup \\ \{(\!|A\ [A \to \textbf{.}AA]\ \sigma\ [A \to A\textbf{.}A]\ \sigma'\ [A \to AA\textbf{.}]\ A|\!) \mid \sigma, \sigma' \in D\}. \quad \square \end{array}$$

Proof sketch of Theorem 34. We apply Theorem 30. By def. $\alpha^{\hat{\delta}}$, we have $\alpha^{\hat{\delta}}(\vdash \xrightarrow{(\!|A} T \xrightarrow{A|\!)} \dashv) = (\!|A\ \alpha^{\hat{\delta}}(T)\ A|\!)$. To get (23), it remains to define $\hat{\mathsf{F}}^{\hat{\delta}}$ such that

$$\alpha^{\hat{\delta}} \circ \hat{\mathsf{F}}^{\hat{d}}_{\textbf{.}}[A \to \sigma\textbf{.}\sigma'] = \hat{\mathsf{F}}^{\hat{\delta}}_{\textbf{.}}[A \to \sigma\textbf{.}\sigma'] \circ \alpha^{\hat{\delta}}. \tag{28}$$

We proceed by structural induction on the length of σ' in $[A \to \sigma\textbf{.}\sigma']$. We let $T \subseteq \mathbf{lfp}^{\subseteq}\ \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!]$ so that T is a set of derivations. We prove (28) for T, by case analysis on the prefix of σ'. This implies the commutation property $\alpha^{\hat{\delta}} \circ \hat{\mathsf{F}}^{\hat{d}}[\![\mathcal{G}]\!](T) = \hat{\mathsf{F}}^{\hat{\delta}}[\![\mathcal{G}]\!] \circ \alpha^{\hat{\delta}}(T)$ for sets T of derivations so that we conclude by Corollary 106. $\square$

Lemma 36. *For all $[A \to \sigma\textbf{.}\sigma'] \in \mathscr{R}^{\textbf{.}}$, $\hat{\mathsf{F}}^{\hat{\delta}}_{\textbf{.}}[A \to \sigma\textbf{.}\sigma'] \in \wp(\hat{\mathcal{D}}) \mapsto \wp(\hat{\mathcal{D}})$ is upper continuous.* $\square$

Proof. By Lemma 28, observing that, given an increasing chain D_i, $i \in \mathbb{N}$ of elements of $\wp(\hat{\mathcal{D}})$, we have $(\!|A\bigcup_{i\in\mathbb{N}} D_i A|\!) = \bigcup_{i\in\mathbb{N}}(\!|AD_iA|\!)$ so $A^{\hat{\delta}}$ is continuous, $\S^{\hat{\delta}}$, which is concatenation $\circ^{\hat{\delta}}$, is continuous, and $[A \to \sigma\textbf{.}B\sigma']\ \bigcup_{i\in\mathbb{N}} D_i.B = [A \to \sigma\textbf{.}B\sigma']\ \bigcup_{i\in\mathbb{N}} D_i.B \wr\, def.\ selection\textbf{.}.B \wr = \bigcup_{i\in\mathbb{N}}[A \to \sigma\textbf{.}B\sigma']\ D_i.B$ by continuity of concatenation, whence $[A \to \sigma\textbf{.}B\sigma']^{\hat{\delta}}$ is continuous in its first argument. $\square$

15.2. Fixpoint bottom-up syntax tree semantics

15.2.1. Syntax tree semantics

The *syntax tree semantics* $\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!] \in \wp(\hat{\mathcal{T}})$ of a context-free grammar $\mathcal{G} = \langle\mathcal{T},\ \mathcal{N},\ \overline{S},\ \mathscr{R}\rangle$ is the set of syntax trees generated by the grammar $\mathcal{G}$ for each nonterminal. It is defined as the syntax tree abstraction of derivation tree semantics, as follows

$$\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!] \quad \triangleq \quad \alpha^{\hat{s}}(\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!]). \tag{29}$$

Lemma 37. $\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!] \in \mathbb{P}_{\mathscr{P},\mathcal{T}}$. $\square$

Proof. By Lemma 33 and definition of $\alpha^{\hat{s}}$. $\square$

15.2.2. Fixpoint bottom-up structural protolanguage semantics

Let the transformer $\hat{\mathsf{F}}^{\hat{s}}[\![\mathcal{G}]\!] \in \wp(\hat{\mathcal{T}}) \mapsto \wp(\hat{\mathcal{T}})$ be defined as follows

$$\begin{aligned} \hat{\mathsf{F}}^{\hat{s}}[\![\mathcal{G}]\!] \quad &\triangleq \quad \boldsymbol{\lambda} S \bullet \bigcup_{A \to \sigma \in \mathscr{R}} (\!|A\ \hat{\mathsf{F}}^{\hat{s}}_{\textbf{.}}[A \to \textbf{.}\sigma]S\ A|\!) \qquad (30)\\ \hat{\mathsf{F}}^{\hat{s}}_{\textbf{.}}[A \to \sigma\textbf{.}a\sigma'] \quad &\triangleq \quad \boldsymbol{\lambda} S \bullet a\ \hat{\mathsf{F}}^{\hat{s}}_{\textbf{.}}[A \to \sigma a\textbf{.}\sigma']S \\ \hat{\mathsf{F}}^{\hat{s}}_{\textbf{.}}[A \to \sigma\textbf{.}B\sigma'] \quad &\triangleq \quad \boldsymbol{\lambda} S \bullet S.B\ \hat{\mathsf{F}}^{\hat{s}}_{\textbf{.}}[A \to \sigma B\textbf{.}\sigma']S \\ \hat{\mathsf{F}}^{\hat{s}}_{\textbf{.}}[A \to \sigma\textbf{.}] \quad &\triangleq \quad \boldsymbol{\lambda} S \bullet \epsilon. \end{aligned}$$

The syntax tree semantics of a grammar $\mathcal{G}$ can be expressed in fixpoint form for transformer $\hat{\mathsf{F}}^{\hat{s}}[\![\mathcal{G}]\!]$ as follows

Theorem 38.

$$\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!] \quad = \quad \boldsymbol{lfp}^{\subseteq}\ \hat{\mathsf{F}}^{\hat{s}}[\![\mathcal{G}]\!]. \quad \square$$

Example 39. For the grammar $\langle\{a\},\ \{A\},\ A,\ \{A \to A, A \to a\}\rangle$, the above syntax tree semantics is the least fixpoint of the equation

$$S \quad = \quad \{(\!|A\ a\ A|\!)\} \cup \{(\!|A\ \sigma\ A|\!) \mid \sigma \in S\}.$$

The iterates (as defined in Appendix A.1) are

$$
\begin{array}{lcl}
S^0 & = & \varnothing \\
S^1 & = & \{(\!|A\ a\ A|\!)\} \\
S^2 & = & \{(\!|A\ a\ A|\!),\ (\!|A\ (\!|A\ a\ A|\!)\ A|\!)\} \\
\ldots & & \ldots \\
S^n & = & \{(\!|A^k\ a\ A|\!)^k \mid 1 \leqslant k \leqslant n\} \\
\ldots & & \ldots \\
S^\omega & = & \bigcup\limits_{n \geq 0} S^n \quad = \quad \{(\!|A^n\ a\ A|\!)^n \mid n \geqslant 1\} \quad = \quad \left\{ \begin{array}{c} A \\ | \\ a \end{array},\ \begin{array}{c} A \\ | \\ A \\ | \\ a \end{array},\ \ldots,\ \begin{array}{c} A \\ | \\ A \\ \vdots \\ A \\ | \\ a \end{array},\ \ldots \right\} \quad \square
\end{array}
$$

Proof of Theorem 38. We apply Corollary 32 and prove (24). For $T, T' \in \wp(\hat{\mathscr{T}})$, we have, by definition of $\alpha^{\hat{s}}$, $\alpha^{\hat{s}}(\vdash \xrightarrow{(\!|A} T \xrightarrow{A|\!)} \dashv) = (\!|A\ \alpha^{\hat{s}}(T)\ A|\!)$, $\alpha^{\hat{s}}([A \to \sigma \centerdot a\sigma']\ a) = a$, $\alpha^{\hat{s}}(T\ T') = \alpha^{\hat{s}}(T)\alpha^{\hat{s}}(T')$, $\alpha^{\hat{s}}([A \to \sigma \centerdot B\sigma']\ D.B = \alpha^{\hat{s}}(D.B) = \alpha^{\hat{s}}(D).B$, by def. selection, and $\alpha^{\hat{s}}([A \to \sigma \centerdot]) = \epsilon$. $\square$

Lemma 40. *For all $[A \to \sigma \centerdot \sigma'] \in \mathscr{R}^{\centerdot}$, $\hat{\mathsf{F}}^{\hat{s}}_{\centerdot}[A \to \sigma \centerdot \sigma'] \in \wp(\hat{\mathscr{T}}) \mapsto \wp(\hat{\mathscr{T}})$ is upper continuous.* $\square$

Proof. By Lemma 28, since concatenation $\circ^{\hat{s}}$ is continuous and given an increasing chain S_i, $i \in \mathbb{N}$ of elements of $\wp(\hat{\mathscr{T}})$, we have $a\ (\bigcup_{i \in \mathbb{N}} S_i) = \bigcup_{i \in \mathbb{N}} (a\ S_i)$ by continuity of concatenation so that $A^{\hat{s}}$ is continuous, $(\bigcup_{i \in \mathbb{N}} S_i).B = \bigcup_{i \in \mathbb{N}} (S_i.B)$ by def. selection $\centerdot.B$ proving that $[A \to \sigma \centerdot B\sigma']^{\hat{s}}$ is continuous. $\square$

15.3. Fixpoint bottom-up protolanguage semantics

15.3.1. Protolanguage semantics

We define the *protolanguage semantics* $\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \in \mathscr{N} \mapsto \wp(\mathscr{V}^\star)$ of context-free grammars $\mathcal{G}$ as the abstraction of their syntax tree semantics, as follows

$$\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \quad \triangleq \quad \dot{\alpha}^{\hat{L}}(\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!]). \tag{31}$$

15.3.2. Fixpoint bottom-up structural protolanguage semantics

We define the protolanguage transformer[5]

$$
\begin{array}{rcl}
\hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!] & \triangleq & \boldsymbol{\lambda}\,\rho \bullet \boldsymbol{\lambda}\,A \bullet \bigcup\limits_{A \to \sigma \in \mathscr{R}} \{A\} \cup \hat{\mathsf{F}}^{\hat{L}}_{\centerdot}[A \to \centerdot\sigma]\rho \\
\hat{\mathsf{F}}^{\hat{L}}_{\centerdot}[A \to \sigma \centerdot a\sigma'] & \triangleq & \boldsymbol{\lambda}\,\rho \bullet a\ \hat{\mathsf{F}}^{\hat{L}}_{\centerdot}[A \to \sigma a \centerdot \sigma']\rho \\
\hat{\mathsf{F}}^{\hat{L}}_{\centerdot}[A \to \sigma \centerdot B\sigma'] & \triangleq & \boldsymbol{\lambda}\,\rho \bullet (\{B\} \cup \rho(B))\ \hat{\mathsf{F}}^{\hat{L}}_{\centerdot}[A \to \sigma B \centerdot \sigma']\rho \\
\hat{\mathsf{F}}^{\hat{L}}_{\centerdot}[A \to \sigma \centerdot] & \triangleq & \boldsymbol{\lambda}\,\rho \bullet \epsilon
\end{array}
\tag{32}
$$

so as to characterize the protolanguage generated by each nonterminal of the grammar $\mathcal{G}$ in fixpoint form,

Theorem 41.

$$\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \quad = \quad \boldsymbol{lfp}^{\subseteq}\ \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!]. \quad \square$$

[5] Recall that $\bigcup_{x \in \varnothing} f(x) = \varnothing$ so that the protolanguage for a nonterminal with no production is empty.

Example 42. If, for the grammar $\langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$, we abstract away in the fixpoint equation of Example 35 the syntax trees for the nonterminal A by the tips of their subtrees, we get the prototype language equation

$$\mathcal{X} \quad = \quad \{A\} \cup \{a\} \cup \mathcal{X}\mathcal{X}.$$

This fixpoint equation is $\rho = \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!](\rho)$ or equivalently $\rho(A) = \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!](\rho)(A)$ that is $\rho(A) = \{A\} \cup \{a\} \cup \rho(A)\rho(A)$, which is $\mathcal{X} = \{A\} \cup \{a\} \cup \mathcal{X}\mathcal{X}$ where $\mathcal{X} \triangleq \rho(A)$. $\square$

Proof sketch of Theorem 41. By Corollary 32 since by def. of $\dot{\alpha}^{\hat{L}}$ in Section 13.3.3, we have $\dot{\alpha}^{\hat{L}}(A^{\natural}(S)) = \dot{\alpha}^{\hat{L}}((\!|ASA|\!)) = \boldsymbol{\lambda}\, B \bullet (\![B = A \mathrel{?} \alpha^{\hat{L}}((\!|ASA|\!).B) \mathrel{⁏} \varnothing]\!) = \boldsymbol{\lambda}\, B \bullet (\![B = A \mathrel{?} \alpha^{\hat{L}}((\!|ASA|\!)) \mathrel{⁏} \varnothing]\!) = \boldsymbol{\lambda}\, B \bullet (\![B = A \mathrel{?} \{A\} \cup \alpha^{\hat{L}}(S) \mathrel{⁏} \varnothing]\!) = A^{\sharp}(\alpha^{\hat{L}}(S))$. It remains to define $\hat{\mathsf{F}}^{\hat{L}}$ such that

$$\alpha^{\hat{L}} \circ \hat{\mathsf{F}}^{\hat{s}}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma'] \quad = \quad \hat{\mathsf{F}}^{\hat{L}}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma'] \circ \dot{\alpha}^{\hat{L}}. \tag{33}$$

We proceed by structural induction on the length of σ' in $[A \to \sigma\boldsymbol{.}\sigma']$ and case analysis on the prefix of σ'. Having proved the commutation property $\dot{\alpha}^{\hat{L}} \circ \hat{\mathsf{F}}^{\hat{s}}[\![\mathcal{G}]\!] = \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!] \circ \dot{\alpha}^{\hat{L}}$, we conclude by Corollary 106. $\square$

Lemma 43. *For all $[A \to \sigma\boldsymbol{.}\sigma'] \in \mathscr{R}$, $\hat{\mathsf{F}}^{\hat{L}}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma'] \in \wp(\mathscr{V}^\star) \mapsto \wp(\mathscr{V}^\star)$ is upper continuous.* $\square$

Proof. By Lemma 28 since $A^{\hat{L}} = \boldsymbol{\lambda}\, L \bullet \boldsymbol{\lambda}\, A' \bullet (\![A' = A \mathrel{?} \{A\} \cup L \mathrel{⁏} \varnothing]\!)$ is pointwise continuous, the junction $\mathrel{⁏}^{\hat{L}}$, which is concatenation $\circ^{\hat{L}}$, is continuous, and $[A \to \sigma\boldsymbol{.}B\sigma']^{\hat{L}} = \boldsymbol{\lambda}\, \rho \bullet \{B\} \cup \rho(B)$ is continuous. $\square$

15.4. Fixpoint bottom-up terminal language semantics

15.4.1. Terminal language semantics

We define the *terminal language semantics* $\mathsf{S}^{\ell}[\![\mathcal{G}]\!] \in \mathscr{N} \mapsto \wp(\mathscr{T}^\star)$ of context-free grammars $\mathcal{G}$ by abstraction of their protolanguage semantics, as follows

$$\mathsf{S}^{\ell}[\![\mathcal{G}]\!] \quad \triangleq \quad \dot{\alpha}^{\ell}(\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!]). \tag{34}$$

15.4.2. Fixpoint right bottom-up structural terminal language semantics

In order to get the classical equational definition of the language generated by a grammar [29,30], let us define the language right transformer

$$\begin{aligned}
\hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!] \quad &\triangleq \quad \boldsymbol{\lambda}\, \rho \bullet \boldsymbol{\lambda}\, A \bullet \bigcup_{A \to \sigma \in \mathscr{R}} \hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \boldsymbol{.}\sigma]\rho \qquad (35)\\
\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\boldsymbol{.}a\sigma'] \quad &\triangleq \quad \boldsymbol{\lambda}\, \rho \bullet a\, \hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma a\boldsymbol{.}\sigma']\rho\\
\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\boldsymbol{.}B\sigma'] \quad &\triangleq \quad \boldsymbol{\lambda}\, \rho \bullet \rho(B)\, \hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma B\boldsymbol{.}\sigma']\rho\\
\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\boldsymbol{.}] \quad &\triangleq \quad \boldsymbol{\lambda}\, \rho \bullet \epsilon.
\end{aligned}$$

We call $\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma']$ the right transformer because it describes the derivation of σ', on the right of the dot. So it is defined by induction on the grammar rule right-hand side from left to right.

The language generated by each nonterminal of the grammar $\mathcal{G}$ can be characterized in fixpoint form, as follows

Theorem 44 (*Ginsburg, Rice, Schützenberger*)**.**

$$\mathsf{S}^{\ell}[\![\mathcal{G}]\!] \quad = \quad \textbf{\textit{lfp}}^{\subseteq}\, \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!]. \quad \square$$

Example 45. If, for the grammar $\mathcal{G} = \langle\{a\},\ \{A\},\ A,\ \{A \to AA, A \to a\}\rangle$, we abstract away the nonterminals in the fixpoint equation of Example 42, we get the language equation

$$\mathcal{X} \quad = \quad \{a\} \cup \mathcal{X}\mathcal{X},$$

which least solution is, according to the Ginsburg–Rice/Chomsky–Schützenberger theorem [31,29,30], the language defined by $\mathcal{G}$. By defining $\mathcal{X} \triangleq \rho(A)$, this is $\rho(A) = \{a\} \cup \rho(A)\rho(A)$ or equivalently $\rho(A) = \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!](\rho)(A)$, that is $\rho = \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!](\rho)$. $\square$

Proof of Theorem 44. By Corollary 32, proving the local soundness and completeness conditions (24). In particular, by def. of $\dot{\alpha}^{\ell}$ and α^{ℓ}, $\dot{\alpha}^{\ell}(\boldsymbol{\lambda}\, A' \bullet (\![A' = A \mathrel{?} \{A\} \cup L \mathrel{⁏} \varnothing]\!)) = \boldsymbol{\lambda}\, A' \bullet (\![A' = A \mathrel{?} \alpha^{\ell}(\{A\} \cup L) \mathrel{⁏} \alpha^{\ell}(\varnothing)]\!) = \boldsymbol{\lambda}\, A' \bullet (\![A' = A \mathrel{?} \alpha^{\ell}(L) \mathrel{⁏} \varnothing]\!)$ and $\alpha^{\ell}(\{B\} \cup \rho(B)) = \alpha^{\ell}(\rho(B)) = \dot{\alpha}^{\ell}(\rho)B$. $\square$

Lemma 46. *For all $[A \to \sigma\boldsymbol{.}\sigma'] \in \mathscr{R}$, $\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma'] \in \wp(\mathscr{T}^\star) \mapsto \wp(\mathscr{T}^\star)$ is upper continuous.* $\square$

Proof. According to Theorem 109, by continuity of $\hat{\mathsf{F}}^{\hat{L}}$ (Lemma 43), commutation $\alpha^{\ell} \circ \hat{\mathsf{F}}^{\hat{L}}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma'] = \hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\boldsymbol{.}\sigma'] \circ \dot{\alpha}^{\ell}$ (25), and α^{ℓ} is onto in $\langle\wp(\mathscr{V}^\star),\ \subseteq\rangle \underset{\alpha^{\ell}}{\overset{\gamma^{\ell}}{\leftrightarrows}} \langle\wp(\mathscr{T}^\star),\ \subseteq\rangle$. $\square$

Abstract Se— mantics $\mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!]$	Protoderivation $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]$	Protoderivation tree $\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!]$	Protosyntax tree $\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!]$	Protolanguage $\mathsf{S}^{\check{L}}[\![\mathcal{G}]\!]$
$\check{\mathsf{D}}^{\check{\sharp}}$	$\wp(\Pi)$	$\wp(\check{\mathcal{D}})$	$\wp(\check{\mathcal{T}})$	$\wp(\mathcal{V}^\star)$
$\sqsubseteq$	$\subseteq$	$\subseteq$	$\subseteq$	$\subseteq$
$\bot$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$\sqcup$	$\cup$	$\cup$	$\cup$	$\cup$
$A^{\check{\sharp}}[\![\mathcal{G}]\!]$	$\{\vdash \xrightarrow{\boxed{A}} \dashv\}$	$\{\boxed{A}\}$	$\{\boxed{A}\}$	$\{A\}$
$\check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!]\phi(A)$	$\mathrm{post}[\boxed{\check{D}}\!\Longmapsto_{\mathcal{G}}]$	$\mathrm{post}[\boxed{\check{\delta}}\!\Longmapsto_{\mathcal{G}}]$	$\mathrm{post}[\boxed{\check{s}}\!\Longmapsto_{\mathcal{G}}]$	$\mathrm{post}[\Longmapsto_{\mathcal{G}}]$

Fig. 4. Semantic instances of the abstract top-down grammar semantics (36).

Abstract Se— mantics $\mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!]$	Follow semantics $\mathsf{S}^{f}[\![\mathcal{G}]\!]$	Accessibility semantics $\mathsf{S}^{a}[\![\mathcal{G}]\!]$
$\check{\mathsf{D}}^{\check{\sharp}}$	$\wp(\mathcal{T} \cup \{\dashv\})$	$\mathbb{B}$
$\sqsubseteq$	$\subseteq$	$\Longrightarrow$
$\bot$	$\emptyset$	ff
$\sqcup$	$\cup$	$\vee$
$A^{\check{\sharp}}[\![\mathcal{G}]\!]$	$\{\dashv \mid A = \bar{S}\}$	$(A = \bar{S})$
$\check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!]\phi(A)$	$\displaystyle\bigcup_{B\to\sigma A\sigma'\in\mathcal{R}} \begin{array}{l}(\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\sigma') \setminus \{\epsilon\}) \cup \\ (\!\!\mid \epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\sigma') \;?\; \phi(B) \;:\; \emptyset \mid\!\!) \end{array}$	$\displaystyle\bigvee_{B\to\sigma A\sigma'\in\mathcal{R}} \phi(B)$

Fig. 5. Flow analysis instances of the abstract top-down grammar semantics (36).

16. Fixpoint top-down abstract semantics

16.1. Top-down abstract interpreter

All top-down semantics $\mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \check{\mathsf{D}}^{\check{\sharp}}$ of context-free grammars $\mathcal{G}$ in the hierarchy of Section 13 are instances of the following abstract interpreter (which generalizes the top-down grammar flow analysis of [8, Def. 8.2.19]).

$$\mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!] \quad = \quad \mathbf{lfp}^{\ddot{\sqsubseteq}}\, \check{\mathsf{F}}^{\check{\sharp}}[\![\mathcal{G}]\!] \quad \text{where} \quad \check{\mathsf{F}}^{\check{\sharp}}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda}\,\phi \bullet \boldsymbol{\lambda}\,A \bullet A^{\check{\sharp}}[\![\mathcal{G}]\!] \sqcup \check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!]\phi(A) \tag{36}$$

and $\langle \check{\mathsf{D}}^{\check{\sharp}},\ \sqsubseteq,\ \bot,\ \sqcup\rangle$ is a cpo/complete lattice extended pointwise to $\langle \mathcal{N} \mapsto \check{\mathsf{D}}^{\check{\sharp}},\ \dot{\sqsubseteq},\ \dot{\bot},\ \dot{\sqcup}\rangle$ and $\langle (\mathcal{N} \mapsto \check{\mathsf{D}}^{\check{\sharp}}) \mapsto (\mathcal{N} \mapsto \check{\mathsf{D}}^{\check{\sharp}}),\ \ddot{\sqsubseteq},\ \ddot{\bot},\ \ddot{\sqcup}\rangle$, the abstract seed is $A^{\check{\sharp}}[\![\mathcal{G}]\!] \in \check{\mathsf{D}}^{\check{\sharp}}$, and the top-down post-transformer is $\check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!] \in \check{\mathsf{D}}^{\check{\sharp}} \mapsto \check{\mathsf{D}}^{\check{\sharp}}$.

16.2. Well-definedness of the top-down abstract interpreter

The existence of the least fixpoint (36) is guaranteed by the following

Hypothesis 47. $\check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!]$ is upper continuous for the ordering $\dot{\sqsubseteq}$ on $\mathcal{N} \mapsto \check{\mathsf{D}}^{\check{\sharp}}$.[6] □

16.3. Instances of the top-down abstract interpreter

The hierarchy of semantics discussed in Section 13 is obtained by the instances of the top-down abstract semantics (36) given in Fig. 4 (post[τ] preserves $\cup$ whence is upper continuous). Observe that by Theorem 17, the maximal protoderivation semantics $\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]$ is of the form (36) for $\check{\mathsf{F}}^{\check{D}}[\![\mathcal{G}]\!]$ is given in Fig. 4. The study of the other instances of the top-down abstract interpreter is forthcoming, in Section 17 for top-down grammar semantics and in Section 20 for top-down grammar analysis.

Classical top-down flow analyses also have the same form given in Fig. 5.

6 Indeed monotony is sufficient [28].

16.4. Soundness of the top-down abstract interpreter

We can define the soundness of an abstract top-down interpreter $\mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!]$ with respect to a concrete interpreter $\mathsf{S}^{\check{\natural}}[\![\mathcal{G}]\!]$ as $\dot{\alpha}(\mathsf{S}^{\check{\natural}}[\![\mathcal{G}]\!]) \sqsubseteq \mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!]$ where $\sqsubseteq$ denotes either $\sqsubseteq$, $=$ or $\sqsupseteq$ and $\langle \hat{\mathsf{D}}^{\hat{\natural}},\ \sqsubseteq^{\natural}\rangle \xleftrightarrows[\alpha]{\gamma} \langle L^{\check{\sharp}},\ \sqsubseteq^{\check{\sharp}}\rangle$ is a Galois connection extended pointwise to $\langle \mathcal{N} \mapsto \hat{\mathsf{D}}^{\hat{\natural}},\ \dot{\sqsubseteq}^{\natural}\rangle \xleftrightarrows[\dot{\alpha}]{\dot{\gamma}} \langle \mathcal{N} \mapsto L^{\check{\sharp}},\ \dot{\sqsubseteq}^{\check{\sharp}}\rangle$. Then the sufficient soundness condition given in Corollary 101 in the form of the commutation condition $\forall \delta \in \mathbb{O} : \dot{\alpha} \circ \check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!](F^{\delta}) \sqsubseteq \check{\mathsf{F}}^{\check{\sharp}}[\![\mathcal{G}]\!] \circ \dot{\alpha}(F^{\delta})$ is implied by the following *local soundness conditions* on the abstract operators

$$\alpha(A^{\check{\natural}}[\![\mathcal{G}]\!]) \quad \sqsubseteq \quad A^{\check{\sharp}}[\![\mathcal{G}]\!] \qquad \text{and} \qquad \dot{\alpha} \circ \check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!] \quad \sqsubseteq \quad \check{\mathsf{T}}^{\check{\natural}}[\![\mathcal{G}]\!] \circ \dot{\alpha}.$$

Note 48. By Corollary 101, the condition can be restricted to $\dot{\alpha} \circ \check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!](\phi) \sqsubseteq \check{\mathsf{T}}^{\check{\natural}}[\![\mathcal{G}]\!] \circ \dot{\alpha}(\phi)$ where ϕ is an iterate of $\check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!]$, or, by Corollary 106 when $\sqsubseteq$ is $=$, we can assume that $\phi \sqsubseteq^{\natural} \mathbf{lfp}^{\sqsubseteq} \check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!]$. $\square$

Theorem 49. *The above local soundness conditions imply the soundness (and completeness whenever $\sqsubseteq$ is $=$) of the abstract top-down interpreter* $\dot{\alpha}(\mathsf{S}^{\check{\natural}}[\![\mathcal{G}]\!]) = \dot{\alpha}(\boldsymbol{lfp}^{\sqsubseteq^{\natural}}\ \check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!]) \sqsubseteq \boldsymbol{lfp}^{\sqsubseteq^{\check{\sharp}}}\ \check{\mathsf{F}}^{\check{\sharp}}[\![\mathcal{G}]\!] = \mathsf{S}^{\check{\sharp}}[\![\mathcal{G}]\!]$. $\square$

Proof. We apply Corollary 101, proving the commutation property

$$\begin{aligned}
& \dot{\alpha} \circ \check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!](\phi)A \quad = \quad \alpha(\check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!](\phi)A) && \text{\{def. } \circ \text{ and pointwise def. } \dot{\alpha}\text{\}} \\
= \quad & \alpha(A^{\check{\natural}}[\![\mathcal{G}]\!]) \sqcup \alpha(\check{\mathsf{T}}^{\check{\natural}}[\![\mathcal{G}]\!]\phi(A)) \\
& && \text{\{def. (36) of } \check{\mathsf{F}}^{\check{\natural}}[\![\mathcal{G}]\!] \text{ and lower adjoint of Galois connection preserves lubs\}} \\
\sqsubseteq \quad & A^{\check{\sharp}}[\![\mathcal{G}]\!] \sqcup \check{\mathsf{T}}^{\check{\sharp}}[\![\mathcal{G}]\!]\dot{\alpha}(\phi)(A) \\
& && \text{\{local soundness conditions, } \sqcup \text{ is } \sqsubseteq\text{-monotonic, and pointwise def. } \dot{\alpha}\text{\}} \\
= \quad & (\check{\mathsf{F}}^{\check{\sharp}}[\![\mathcal{G}]\!] \circ \dot{\alpha})(\phi)A && \text{\{def. (36) of } \check{\mathsf{F}}^{\check{\sharp}}[\![\mathcal{G}]\!] \text{ and } \circ\text{\}}. \quad \square
\end{aligned}$$

17. The hierarchy of top-down grammar semantics

17.1. Fixpoint top-down protoderivation tree semantics

17.1.1. Protoderivation tree semantics

The *protoderivation tree semantics* $\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \wp(\check{\mathcal{D}})$ of a context-free grammar $\mathcal{G} = \langle \mathcal{T},\ \mathcal{N},\ \overline{S},\ \mathcal{R}\rangle$, is the set of protoderivation trees generated by the grammar $\mathcal{G}$. It is defined as the protoderivation tree abstraction of the protoderivation semantics, as follows

$$\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!] \quad \triangleq \quad \alpha^{\check{\delta}}(\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]). \tag{37}$$

Lemma 50. $\forall A \in \mathcal{N} : \mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!](A) \in \mathbb{P}_{\mathcal{P},\check{\mathcal{U}}}$. $\square$

Proof. By Lemma 5 and definition of $\alpha^{\check{\delta}}$. $\square$

17.1.2. Protoderivation tree derivation

Let us define $\check{\mathsf{R}}^{\check{\delta}} \in \mathcal{R} \mapsto \check{\mathcal{D}}$ as

$$\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma] \quad \triangleq \quad (\!| A\ \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \bullet\sigma]\ A |\!) \tag{38}$$

where $\check{\mathsf{R}}^{\check{\delta}}_{\bullet} \in \mathcal{R}^{\bullet} \mapsto \check{\mathcal{D}}$ is

$$\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma \bullet a\sigma'] \quad \triangleq \quad [A \to \sigma \bullet a\sigma']\ a\ \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma a \bullet \sigma'] \tag{39}$$

$$\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma \bullet B\sigma'] \quad \triangleq \quad [A \to \sigma \bullet B\sigma']\ \boxed{B}\ \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma B \bullet \sigma'] \tag{40}$$

$$\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma \bullet] \quad \triangleq \quad [A \to \sigma \bullet] \tag{41}$$

so that

$$\begin{aligned}
&\check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}} \check{\delta}' && (42)\\
\triangleq\ & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \check{\delta} = \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \wedge \\
& \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \check{\delta}' = \varsigma_1 \check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1] \varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n] \varsigma_{n+1}.
\end{aligned}$$

17.1.3. Fixpoint top-down protoderivation tree semantics

Theorem 51.

$$\begin{aligned}
\mathsf{S}^{\check{\delta}}[\![\mathscr{G}]\!] &= \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^{\check{\delta}}[\![\mathscr{G}]\!] \\
\textit{where}\quad \check{\mathsf{F}}^{\check{\delta}}[\![\mathscr{G}]\!] &\triangleq \boldsymbol{\lambda}\,\phi \bullet \boldsymbol{\lambda}\,A \bullet \{\boxed{A}\} \cup \operatorname{post}[\overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}}](\phi(A)). \quad \square
\end{aligned}$$

Proof. We apply Theorem 49. In the proof, we assume that ϕ is an iterate of $\check{\mathsf{F}}^{\check{D}}[\![\mathscr{G}]\!]$ whence, by (17), $\phi(A) = \operatorname{post}[\overset{\boxed{\check{D}}\ n*}{\Longrightarrow}_{\mathscr{G}}](\{\vdash \xrightarrow{\boxed{A}} \dashv\})$, as shown in Example 107. Let us calculate

— $\alpha^{\check{\delta}}(\boldsymbol{\lambda}\,A \bullet \{\vdash \xrightarrow{\boxed{A}} \dashv\}) \quad = \quad \boldsymbol{\lambda}\,A \bullet \{\boxed{A}\}$ $\quad$ ≀def. $\alpha^{\check{\delta}}$≀

— $\alpha^{\check{\delta}}(\boldsymbol{\lambda}\,A \bullet \operatorname{post}[\overset{\boxed{\check{D}}}{\Longrightarrow}_{\mathscr{G}}]\phi(A)) \quad = \quad \boldsymbol{\lambda}\,A \bullet \{\check{\delta}' \mid \exists \check{\delta} \in \alpha^{\check{\delta}}(\phi(A)) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}} \check{\delta}'\}$
≀def. post and $\alpha^{\check{\delta}}$, provided we can define $\overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}}$ such that $\{\alpha^{\check{\delta}}(\pi') \mid \exists \pi \in \phi(A) : \pi \overset{\boxed{\check{D}}}{\Longrightarrow}_{\mathscr{G}} \pi'\} = \{\check{\delta}' \mid \exists \check{\delta} \in \alpha^{\check{\delta}}(\phi(A)) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}} \check{\delta}'\}$≀

$=\quad \boldsymbol{\lambda}\,A \bullet \operatorname{post}[\overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}}](\alpha^{\check{\delta}}(\phi)(A))$ $\quad$ ≀def. post and $\alpha^{\check{\delta}}$≀.

— $\alpha^{\check{\delta}}(\check{\mathsf{R}}^{\check{D}}[A \to \sigma]) \quad = \quad \alpha^{\check{\delta}}(\vdash \xrightarrow{(\!|A} \check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \bullet\sigma] \xrightarrow{A|\!)} \dashv)$ $\quad$ ≀def. (11) of $\check{\mathsf{R}}^{\check{D}}_{\bullet}$ and $\alpha^{\check{\delta}}$≀

$=\quad (\!|A\ \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \bullet\sigma]\ A|\!)$

by defining $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet\sigma'] \triangleq \alpha^{\check{\delta}}(\check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma\bullet\sigma'])$ by induction on the length $|\sigma'|$ of σ', as follows.

— $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet a\sigma'] \quad = \quad [A \to \sigma\bullet a\sigma']\ a\ \alpha^{\check{\delta}}(\check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma a\bullet\sigma'])$ $\quad$ ≀def. $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}$, (12) of $\check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma\bullet a\sigma']$, and $\alpha^{\check{\delta}}$≀

$=\quad [A \to \sigma\bullet a\sigma']\ a\ \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma a\bullet\sigma']$ $\quad$ ≀ind. def.≀

— $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet B\sigma'] \quad = \quad [A \to \sigma\bullet B\sigma']\ \boxed{B}\ \alpha^{\check{\delta}}(\check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma B\bullet\sigma'])$ $\quad$ ≀def. $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}$, (13) of $\check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma\bullet B\sigma']$, and $\alpha^{\check{\delta}}$≀

$=\quad [A \to \sigma\bullet B\sigma']\ \boxed{B}\ \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma B\bullet\sigma']$ $\quad$ ≀ind. def.≀

— $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet] \quad = \quad [A \to \sigma\bullet]$ $\quad$ ≀def. $\check{\mathsf{R}}^{\check{\delta}}_{\bullet}$, (14) of $\check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma\bullet]$, and $\alpha^{\check{\delta}}$≀.

By induction on $|\sigma'|$, we observe that $\alpha^{\check{\delta}}(\langle\varpi',\ \varpi'\rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma\bullet\sigma']) = \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet\sigma']$. It follows that

— $\alpha^{\check{\delta}}(\langle\varpi,\ \varpi'\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A \to \sigma]) \quad = \quad \alpha^{\check{\delta}}(\varpi)(\!|A\check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet\sigma']A|\!)\alpha^{\check{\delta}}(\varpi')$
≀def. (11) of $\check{\mathsf{R}}^{\check{D}}$, $\alpha^{\check{\delta}}$ and $\langle\varpi',\ \varpi'\rangle \uparrow \bullet$, and since $\alpha^{\check{\delta}}(\langle\varpi',\ \varpi'\rangle \uparrow \check{\mathsf{R}}^{\check{D}}_{\bullet}[A \to \sigma\bullet\sigma']) = \check{\mathsf{R}}^{\check{\delta}}_{\bullet}[A \to \sigma\bullet\sigma']$≀

$=\quad \alpha^{\check{\delta}}(\varpi)\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma]\alpha^{\check{\delta}}(\varpi')$ $\quad$ ≀def. (38) of $\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma]$≀.

Let us examine the pending condition

$$\begin{aligned}
&\{\alpha^{\check{\delta}}(\pi') \mid \exists \pi \in \phi(A) : \pi \overset{\boxed{\check{D}}}{\Longrightarrow}_{\mathscr{G}} \pi'\} \subseteq \{\check{\delta}' \mid \exists \check{\delta} \in \alpha^{\check{\delta}}(\phi(A)) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}} \check{\delta}'\} \\
\Longleftarrow\ & \forall \pi \in \phi(A) : \forall \pi' : (\pi \overset{\boxed{\check{D}}}{\Longrightarrow}_{\mathscr{G}} \pi') \Longrightarrow (\exists \check{\delta} \in \alpha^{\check{\delta}}(\phi(A)) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}} \alpha^{\check{\delta}}(\pi')) && \wr\text{def. } \subseteq \wr \\
\Longleftarrow\ & \forall \pi, \pi' : (\pi \overset{\boxed{\check{D}}}{\Longrightarrow}_{\mathscr{G}} \pi') \Longrightarrow (\alpha^{\check{\delta}}(\pi) \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}} \alpha^{\check{\delta}}(\pi')) && \wr\text{choosing } \check{\delta} = \alpha^{\check{\delta}}(\pi)\wr.
\end{aligned}$$

This sufficient condition leads to the design of $\overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathscr{G}}$ as follows

$$
\begin{array}{ll}
& \pi \;\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\; \pi' \\
\Longrightarrow & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, \varpi_1, \ldots, \varpi_{n+1} \in \mathscr{S}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{\delta}}(\pi) = \\
& \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\varpi_1)\; \boxed{A_1}\; \alpha^{\check{\delta}}(\varpi_2)\alpha^{\check{\delta}}(\varsigma_2) \ldots \alpha^{\check{\delta}}(\varsigma_n)\alpha^{\check{\delta}}(\varpi_n)\; \boxed{A_n}\; \alpha^{\check{\delta}}(\varpi_{n+1})\alpha^{\check{\delta}}(\varsigma_{n+1}) \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \alpha^{\check{\delta}}(\pi') = \\
& \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\langle \varpi_1,\ \varpi_2\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A_1 \to \sigma_1])\alpha^{\check{\delta}}(\varsigma_2) \ldots \alpha^{\check{\delta}}(\varsigma_n)\alpha^{\check{\delta}}(\langle \varpi_n,\ \varpi_{n+1}\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A_n \to \sigma_n])\alpha^{\check{\delta}}(\varsigma_{n+1}) \\
& \hfill \wr\text{def. (15) of } \boxed{D}\!\!\Longrightarrow_{\mathscr{G}}, =, \text{and } \alpha^{\check{\delta}}\wr \\
\Longleftrightarrow & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, \varpi_1, \ldots, \varpi_{n+1} \in \mathscr{S}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{\delta}}(\pi) = \\
& \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\varpi_1)\; \boxed{A_1}\; \alpha^{\check{\delta}}(\varpi_2)\alpha^{\check{\delta}}(\varsigma_2) \ldots \alpha^{\check{\delta}}(\varsigma_n)\alpha^{\check{\delta}}(\varpi_n)\; \boxed{A_n}\; \alpha^{\check{\delta}}(\varpi_{n+1})\alpha^{\check{\delta}}(\varsigma_{n+1}) \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \alpha^{\check{\delta}}(\pi') = \\
& \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\varpi_1)\check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1]\alpha^{\check{\delta}}(\varpi_2)\alpha^{\check{\delta}}(\varsigma_2) \ldots \alpha^{\check{\delta}}(\varsigma_n)\alpha^{\check{\delta}}(\varpi_n)\check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n]\alpha^{\check{\delta}}(\varpi_{n+1})\alpha^{\check{\delta}}(\varsigma_{n+1}) \\
& \hfill \wr\text{since } \alpha^{\check{\delta}}(\langle \varpi,\ \varpi'\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A \to \sigma]) = \alpha^{\check{\delta}}(\varpi)\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma]\alpha^{\check{\delta}}(\varpi')\wr \\
\Longleftrightarrow & \exists n > 0, \varsigma'_1, \ldots, \varsigma'_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{\delta}}(\pi) = \varsigma'_1\; \boxed{A_1}\; \varsigma'_2 \ldots \varsigma'_n\; \boxed{A_n}\; \varsigma'_{n+1} \wedge \forall i \in [1, n] : A_i \to \\
& \sigma_i \in \mathscr{R} \wedge \alpha^{\check{\delta}}(\pi') = \varsigma'_1\check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1]\varsigma'_2 \ldots \varsigma'_n\check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n]\varsigma'_{n+1} \qquad \wr\text{by letting } \varsigma'_i = \alpha^{\check{\delta}}(\varsigma_i)\alpha^{\check{\delta}}(\varpi_i), i = 1, \ldots, n+1\wr \\
\Longleftrightarrow & \alpha^{\check{\delta}}(\pi') \;\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}\; \alpha^{\check{\delta}}(\pi) \hfill \wr\text{by defining } \boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}} \text{ as in (42).}\wr
\end{array}
$$

For the inverse inclusion, we have

$$
\begin{array}{ll}
& \{\check{\delta}' \mid \exists \check{\delta} \in \alpha^{\check{\delta}}(\phi(A)) : \check{\delta} \;\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}\; \check{\delta}'\} \subseteq \{\alpha^{\check{\delta}}(\pi') \mid \exists \pi \in \phi(A) : \pi \;\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\; \pi'\} \\
\Longleftarrow & \forall \pi'' \in \phi(A) : \forall \check{\delta}' : (\alpha^{\check{\delta}}(\pi'') \;\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}\; \check{\delta}') \Longrightarrow (\exists \pi \in \phi(A) : \exists \pi' : \pi \;\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\; \pi' \wedge \check{\delta}' = \alpha^{\check{\delta}}(\pi')) \\
& \hfill \wr\text{def. } \subseteq \text{ and since } \check{\delta} \in \alpha^{\check{\delta}}(\phi(A))\wr \\
\Longleftarrow & \forall \pi'' \in \phi(A) : \forall \check{\delta}' : (\alpha^{\check{\delta}}(\pi'') \;\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}\; \check{\delta}') \Longrightarrow (\exists \pi' : \pi'' \;\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\; \pi' \wedge \check{\delta}' = \alpha^{\check{\delta}}(\pi')) \hfill \wr\text{choosing } \pi = \pi''\wr
\end{array}
$$

We have $\pi'' \in \phi(A)$ so $(\vdash \xrightarrow{\boxed{A}} \dashv) \;\boxed{D}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}}\; \pi''$ hence, by def. (15) of $\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}$, π'' has necessarily the form $\varsigma_1\varpi_1 \xrightarrow{\boxed{A_1}} \varpi_2\varsigma_2 \ldots \varsigma_{m-1}\varpi_{m-1} \xrightarrow{\boxed{A_{m-1}}} \varpi_m\varsigma_m$ where $m \geqslant 0$ ($m = 0$ if π'' has no nonterminal variable). It follows that

$$
\begin{array}{ll}
& \alpha^{\check{\delta}}(\pi'') \;\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}\; \check{\delta}' \\
\Longrightarrow & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, \varpi_1, \ldots, \varpi_{n+1} \in \mathscr{S}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \\
& \pi'' = \varsigma_1\varpi_1 \xrightarrow{\boxed{A_1}} \varpi_2\varsigma_2 \ldots \varsigma_n\varpi_n \xrightarrow{\boxed{A_n}} \varpi_{n+1}\varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \check{\delta}' = \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\varpi_1)\check{\mathsf{R}}^{\check{\delta}}[A_1 \to \\
& \sigma_1]\alpha^{\check{\delta}}(\varpi_2)\alpha^{\check{\delta}}(\varsigma_2) \ldots \alpha^{\check{\delta}}(\varsigma_n)\alpha^{\check{\delta}}(\varpi_n)\check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n]\alpha^{\check{\delta}}(\varpi_{n+1})\alpha^{\check{\delta}}(\varsigma_{n+1}) \\
& \hfill \wr\text{def. (42) of } \boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}} \text{ and def. } \alpha^{\check{\delta}} \text{ so that } \varsigma'_1 = \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\varpi_1), \ldots, \varsigma'_{n+1} = \alpha^{\check{\delta}}(\varpi_{n+1})\alpha^{\check{\delta}}(\varsigma_{n+1})\wr \\
\Longrightarrow & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, \varpi_1, \ldots, \varpi_{n+1} \in \mathscr{S}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \\
& \pi'' = \varsigma_1\varpi_1 \xrightarrow{\boxed{A_1}} \varpi_2\varsigma_2 \ldots \varsigma_n\varpi_n \xrightarrow{\boxed{A_n}} \varpi_{n+1}\varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \check{\delta}' = \alpha^{\check{\delta}}(\varsigma_1)\alpha^{\check{\delta}}(\langle \varpi_1, \\
& \varpi_2\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A_1 \to \sigma_1])\alpha^{\check{\delta}}(\varsigma_2) \ldots \alpha^{\check{\delta}}(\varsigma_n)\alpha^{\check{\delta}}(\langle \varpi_n,\ \varpi_{n+1}\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A_n \to \sigma_n])\alpha^{\check{\delta}}(\varsigma_{n+1}) \hfill \wr\text{since } \alpha^{\check{\delta}}(\langle \varpi, \\
& \varpi'\rangle \uparrow \check{\mathsf{R}}^{\check{D}}[A \to \sigma]) = \alpha^{\check{\delta}}(\varpi)\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma]\alpha^{\check{\delta}}(\varpi')\wr \\
\Longrightarrow & \exists \pi' : \pi'' \;\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\; \pi' \wedge \check{\delta}' = \alpha^{\check{\delta}}(\pi') \hfill \wr\text{def. } \alpha^{\check{\delta}} \text{ and (15) of } \boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\wr. \quad \square
\end{array}
$$

Observe that as a corollary of this proof, we have just shown that

Corollary 52.

$$\{\alpha^{\check{\delta}}(\pi) \mid \exists A \in \mathscr{N} : (\vdash \xrightarrow{\boxed{A}} \dashv) \;\boxed{D}\!\!\Longrightarrow_{\mathscr{G}}\; \pi\} \quad = \quad \{\check{\delta} \mid \exists A \in \mathscr{N} : \boxed{A} \;\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}\; \check{\delta}\}. \quad \square$$

Corollary 53.

$$\mathsf{S}^{\check{\delta}}[\![\mathscr{G}]\!] \quad = \quad \boldsymbol{\lambda} A \bullet \{\check{\delta} \in \check{\mathscr{D}} \mid \boxed{A} \;\boxed{\check{\delta}}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}}\; \check{\delta}\}. \quad \square$$

Proof. By Theorem 51, $\mathsf{S}^{\check{\delta}}[\![\mathscr{G}]\!] = \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^{\check{\delta}}[\![\mathscr{G}]\!]$ where $\check{\mathsf{F}}^{\check{\delta}}[\![\mathscr{G}]\!] = \boldsymbol{\lambda}\phi \bullet \boldsymbol{\lambda} A \bullet \{\boxed{A}\} \cup \mathrm{post}[\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}](\phi(A))$ so $\mathsf{S}^{\check{\delta}}[\![\mathscr{G}]\!](A) = \mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \{\boxed{A}\} \cup \mathrm{post}[\boxed{\check{\delta}}\!\!\Longrightarrow_{\mathscr{G}}]X$ by Example 105 whence $\mathsf{S}^{\check{\delta}}[\![\mathscr{G}]\!](A) = \mathrm{post}[\boxed{\check{\delta}}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}}](\{\boxed{A}\}) = \{\check{\delta} \in \check{\mathscr{D}} \mid \boxed{A} \;\boxed{\check{\delta}}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}}\; \check{\delta}\}$ by (A.1). $\square$

17.2. Fixpoint top-down protosyntax tree semantics

17.2.1. Protosyntax tree semantics

The *protosyntax tree semantics* $\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \wp(\check{\mathcal{T}})$ of a context-free grammar $\mathcal{G} = \langle \mathcal{T}, \mathcal{N}, \bar{S}, \mathcal{R}\rangle$ is the set of protosyntax trees generated by the grammar $\mathcal{G}$ for each nonterminal. It is defined as the protosyntax tree abstraction of the protoderivation tree semantics, as follows

$$\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!] \quad \triangleq \quad \alpha^{\check{s}}(\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!]). \tag{43}$$

17.2.2. Protosyntax tree derivation

Let us define $\check{\mathsf{R}}^{\check{s}} \in \mathcal{R} \mapsto \check{\mathcal{T}}$ such that

$$\check{\mathsf{R}}^{\check{s}}[A \to \sigma] \quad \triangleq \quad (\!| A\, \check{\mathsf{R}}^{\check{s}}_{\centerdot}[A \to \centerdot\sigma]\, A |\!) \tag{44}$$

where $\check{\mathsf{R}}^{\check{s}}_{\centerdot} \in \mathcal{R}^{\centerdot} \mapsto \check{\mathcal{T}}$ is

$$\check{\mathsf{R}}^{\check{s}}_{\centerdot}[A \to \sigma\centerdot a\sigma'] \quad \triangleq \quad a\, \check{\mathsf{R}}^{\check{s}}_{\centerdot}[A \to \sigma a\centerdot\sigma'] \qquad \check{\mathsf{R}}^{\check{s}}_{\centerdot}[A \to \sigma\centerdot B\sigma'] \quad \triangleq \quad \boxed{B}\, \check{\mathsf{R}}^{\check{s}}_{\centerdot}[A \to \sigma B\centerdot\sigma']$$
$$\check{\mathsf{R}}^{\check{s}}_{\centerdot}[A \to \sigma\centerdot] \quad \triangleq \quad \epsilon$$

so that

$$\begin{aligned} &\check{\tau} \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \check{\tau}' \\ \triangleq\ & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathcal{N}, \sigma_1, \ldots, \sigma_n \in \mathcal{V}^\star : \check{\tau} = \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \wedge \\ & \forall i \in [1, n] : A_i \to \sigma_i \in \mathcal{R} \wedge \check{\tau}' = \varsigma_1 \check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1]\varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n]\varsigma_{n+1}. \end{aligned} \tag{45}$$

17.2.3. Fixpoint top-down structural protosyntax tree semantics

Theorem 54.

$$\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!] \quad = \quad \textit{\textbf{lfp}}^{\subseteq}\, \check{\mathsf{F}}^{\check{s}}[\![\mathcal{G}]\!] \quad \textit{where} \quad \check{\mathsf{F}}^{\check{s}}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda}\, \phi \bullet \boldsymbol{\lambda}\, A \bullet \{\boxed{A}\} \cup \mathrm{post}[\overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}}]\phi(A). \quad \square$$

Proof. We apply Theorem 49 where Theorem 51 provides a fixpoint characterization of $\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^{\check{\delta}}[\![\mathcal{G}]\!]$. Given an iterate ϕ of $\check{\mathsf{F}}^{\check{\delta}}[\![\mathcal{G}]\!]$, we have to check the following local soundness and completeness conditions.

— $\alpha^{\check{s}}(\boldsymbol{\lambda}\, A \bullet \{\boxed{A}\}) \quad = \quad \boldsymbol{\lambda}\, A \bullet \{\boxed{A}\}$ $\qquad$ (def. $\alpha^{\check{s}}$)

— $\alpha^{\check{s}}(\boldsymbol{\lambda}\, A \bullet \mathrm{post}[\overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}}]\phi(A)) = \boldsymbol{\lambda}\, A \bullet \{\alpha^{\check{s}}(\check{\delta}') \mid \exists \check{\delta} \in \phi(A) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}'\}$ $\qquad$ (def. post, $\alpha^{\check{s}}$)

$= \quad \boldsymbol{\lambda}\, A \bullet \{\check{\tau}' \mid \exists \check{\tau} \in \alpha^{\check{s}}(\phi(A)) : \check{\tau} \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \check{\tau}'\}$

(provided we can define $\overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}}$ such that $\{\alpha^{\check{s}}(\check{\delta}') \mid \exists \check{\delta} \in \phi(A) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}'\} = \{\check{\tau}' \mid \exists \check{\tau} \in \alpha^{\check{s}}(\phi(A)) : \check{\tau} \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \check{\tau}'\}$)

$= \quad \boldsymbol{\lambda}\, A \bullet \mathrm{post}[\overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}}](\alpha^{\check{s}}(\phi)(A))$ $\qquad$ (def. post and $\alpha^{\check{s}}$.)

The design of $\overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}}$ follows from the evaluation of the condition

$$\begin{aligned} & \{\alpha^{\check{s}}(\check{\delta}') \mid \exists \check{\delta} \in \phi(A) : \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}'\} \subseteq \{\check{\tau}' \mid \exists \check{\tau} \in \alpha^{\check{s}}(\phi(A)) : \check{\tau} \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \check{\tau}'\} \\ \Longleftrightarrow\ & \forall \check{\delta} \in \phi(A) : \forall \check{\delta}' : (\check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}') \Longrightarrow (\exists \check{\tau} \in \alpha^{\check{s}}(\phi(A)) : \check{\tau} \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \alpha^{\check{s}}(\check{\delta}')) && \text{(def. } \subseteq, \exists) \\ \Longleftarrow\ & \forall \check{\delta} \in \phi(A) : \forall \check{\delta}' : (\check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}') \Longrightarrow (\alpha^{\check{s}}(\check{\delta}) \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \alpha^{\check{s}}(\check{\delta}')) && \text{(choosing } \check{\tau} = \alpha^{\check{s}}(\check{\delta})) \end{aligned}$$

as follows

$$\begin{aligned} & \check{\delta} \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}' \\ \Longrightarrow\ & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathcal{N}, \sigma_1, \ldots, \sigma_n \in \mathcal{V}^\star : \alpha^{\check{s}}(\check{\delta}) = \alpha^{\check{s}}(\varsigma_1)\boxed{A_1}\alpha^{\check{s}}(\varsigma_2) \ldots \alpha^{\check{s}}(\varsigma_n)\boxed{A_n}\alpha^{\check{s}}(\varsigma_{n+1}) \wedge \forall i \in \\ & [1, n] : A_i \to \sigma_i \in \mathcal{R} \wedge \alpha^{\check{s}}(\check{\delta}') = \alpha^{\check{s}}(\varsigma_1)\alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1])\alpha^{\check{s}}(\varsigma_2) \ldots \alpha^{\check{s}}(\varsigma_n)\alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n])\alpha^{\check{s}}(\varsigma_{n+1}) \\ & && \text{(def. (42) of } \overset{\boxed{\check{\delta}}}{\Longrightarrow}_{\mathcal{G}}, =, \text{and } \alpha^{\check{s}}) \\ \Longleftrightarrow\ & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathcal{N}, \sigma_1, \ldots, \sigma_n \in \mathcal{V}^\star : \alpha^{\check{s}}(\check{\delta}) = \alpha^{\check{s}}(\varsigma_1)\boxed{A_1}\alpha^{\check{s}}(\varsigma_2) \ldots \alpha^{\check{s}}(\varsigma_n)\boxed{A_n}\alpha^{\check{s}}(\varsigma_{n+1}) \wedge \forall i \in \\ & [1, n] : A_i \to \sigma_i \in \mathcal{R} \wedge \alpha^{\check{s}}(\check{\delta}') = \alpha^{\check{s}}(\varsigma_1)\check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1]\alpha^{\check{s}}(\varsigma_2) \ldots \alpha^{\check{s}}(\varsigma_n)\check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n]\alpha^{\check{s}}(\varsigma_{n+1}) \\ & && \text{(by defining } \check{\mathsf{R}}^{\check{s}} \text{ as in (44) so that } \alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma]) = \check{\mathsf{R}}^{\check{s}}[A \to \sigma]) \\ \Longrightarrow\ & \exists n > 0, \varsigma'_1, \ldots, \varsigma'_{n+1}, A_1, \ldots, A_n \in \mathcal{N}, \sigma_1, \ldots, \sigma_n \in \mathcal{V}^\star : \alpha^{\check{s}}(\check{\delta}) = \varsigma'_1\boxed{A_1}\varsigma'_2 \ldots \varsigma'_n\boxed{A_n}\varsigma'_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \\ & \mathcal{R} \wedge \alpha^{\check{s}}(\check{\delta}') = \varsigma'_1\check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1]\varsigma'_2 \ldots \varsigma'_n\check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n]\varsigma'_{n+1} && \text{(by letting } \varsigma'_i = \alpha^{\check{s}}(\varsigma_i), i = 1, \ldots, n+1) \\ \Longleftrightarrow\ & \alpha^{\check{s}}(\check{\delta}) \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \alpha^{\check{s}}(\check{\delta}') && \text{(by defining } \overset{\boxed{\check{s}}}{\Longrightarrow}_{\mathcal{G}} \text{ as in (45)).} \end{aligned}$$

Inversely, we must also check that

$$
\begin{aligned}
& \{\check{\tau}' \mid \exists \check{\tau} \in \alpha^{\check{s}}(\phi(A)) : \check{\tau} \ \boxed{s}\!\!\Longrightarrow_{\mathscr{G}} \check{\tau}'\} \subseteq \{\alpha^{\check{s}}(\check{\delta}') \mid \exists \check{\delta} \in \phi(A) : \check{\delta} \ \boxed{\delta}\!\!\Longrightarrow_{\mathscr{G}} \check{\delta}'\} \\
\Longleftrightarrow \quad & \forall \check{\delta}'' \in \phi(A) : \forall \check{\tau}' : (\alpha^{\check{s}}(\check{\delta}'') \ \boxed{s}\!\!\Longrightarrow_{\mathscr{G}} \check{\tau}') \Longrightarrow (\exists \check{\delta} \in \phi(A) : \exists \check{\delta}' : \check{\delta} \ \boxed{\delta}\!\!\Longrightarrow_{\mathscr{G}} \check{\delta}' \wedge \check{\tau}' = \alpha^{\check{s}}(\check{\delta}')) \\
& \qquad \wr \text{def. } \subseteq \text{ and since. } \check{\tau} \in \alpha^{\check{s}}(\phi(A)) \text{ so } \exists \check{\delta}'' \in \phi(A) : \check{\tau} = \alpha^{\check{s}}(\check{\delta}'') \wr
\end{aligned}
$$

We have $\check{\delta}'' \in \phi(A)$ and ϕ is an iterate of $\check{\mathsf{F}}^{\check{\delta}}[\![\mathscr{G}]\!]$ hence, $\boxed{\overline{S}} \ \boxed{s}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}} \check{\delta}''$, so by def. (45) of $\boxed{s}\!\!\Longrightarrow_{\mathscr{G}}$, $\check{\delta}''$ has necessarily the form $\varsigma_1 \boxed{A'_1} \varsigma_2 \ldots \varsigma_m \boxed{A'_m} \varsigma_{m+1}$, $m \geqslant 0$.

$$
\begin{aligned}
& \alpha^{\check{s}}(\check{\delta}'') \ \boxed{s}\!\!\Longrightarrow_{\mathscr{G}} \check{\tau}' \\
\Longleftrightarrow \quad & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{s}}(\check{\delta}'') = \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \\
& \mathscr{R} \wedge \check{\tau}' = \varsigma_1 \check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1] \varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n] \varsigma_{n+1} \qquad \wr \text{def. (45) of } \boxed{s}\!\!\Longrightarrow_{\mathscr{G}} \wr \\
\Longleftrightarrow \quad & \exists n > 0, \varsigma'_1, \ldots, \varsigma'_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{s}}(\check{\delta}'') = \alpha^{\check{s}}(\varsigma'_1) \boxed{A_1} \alpha^{\check{s}}(\varsigma'_2) \ldots \alpha^{\check{s}}(\varsigma'_n) \boxed{A_n} \alpha^{\check{s}}(\varsigma'_{n+1}) \wedge \forall i \in \\
& [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \check{\tau}' = \alpha^{\check{s}}(\varsigma'_1) \check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1] \alpha^{\check{s}}(\varsigma'_2) \ldots \alpha^{\check{s}}(\varsigma'_n) \check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n] \alpha^{\check{s}}(\varsigma'_{n+1}) \\
& \qquad \wr \text{def. } \alpha^{\check{s}} \text{ so that } \varsigma_i = \alpha^{\check{s}}(\varsigma'_i), i = 1, \ldots, n+1 \wr \\
\Longleftrightarrow \quad & \exists n > 0, \varsigma'_1, \ldots, \varsigma'_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{s}}(\check{\delta}'') = \alpha^{\check{s}}(\varsigma'_1) \boxed{A_1} \alpha^{\check{s}}(\varsigma'_2) \ldots \alpha^{\check{s}}(\varsigma'_n) \boxed{A_n} \alpha^{\check{s}}(\varsigma'_{n+1}) \wedge \forall i \in \\
& [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \check{\tau}' = \alpha^{\check{s}}(\varsigma'_1) \alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1]) \alpha^{\check{s}}(\varsigma'_2) \ldots \alpha^{\check{s}}(\varsigma'_n) \alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n]) \alpha^{\check{s}}(\varsigma'_{n+1}) \\
& \qquad \wr \text{by def. (44) of } \check{\mathsf{R}}^{\check{s}} \text{ so that } \alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A \to \sigma]) = \check{\mathsf{R}}^{\check{s}}[A \to \sigma] \wr \\
\Longrightarrow \quad & \exists \check{\delta} \in \phi(A) : \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \check{\delta} = \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \wedge \forall i \in [1, n] : \\
& A_i \to \sigma_i \in \mathscr{R} \wedge \check{\tau}' = \alpha^{\check{s}}(\varsigma_1) \alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1]) \alpha^{\check{s}}(\varsigma_2) \ldots \alpha^{\check{s}}(\varsigma_n) \alpha^{\check{s}}(\check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n]) \alpha^{\check{s}}(\varsigma_{n+1}) \\
& \qquad \wr \text{by choosing } \check{\delta} = \check{\delta}'' \text{ which, since } \alpha^{\check{s}}(\check{\delta}'') = \alpha^{\check{s}}(\varsigma'_1) \boxed{A_1} \alpha^{\check{s}}(\varsigma'_2) \ldots \alpha^{\check{s}}(\varsigma'_n) \boxed{A_n} \alpha^{\check{s}}(\varsigma'_{n+1}) \text{ and } \check{\delta}'' = \\
& \qquad \varsigma''_1 \boxed{A'_1} \varsigma''_2 \ldots \varsigma''_m \boxed{A'_m} \varsigma''_{m+1} \text{ so, by def. of } \alpha^{\check{s}}, m \geqslant n \text{ and } \check{\delta}'' \text{ has the form } \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \text{ with } \alpha^{\check{s}}(\varsigma'_i) = \alpha^{\check{s}}(\varsigma_i), \\
& \qquad i = 1, \ldots, n+1 \wr \\
\Longleftrightarrow \quad & \exists \check{\delta} \in \phi(A) : \exists \check{\delta}' : \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \check{\delta} = \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \wedge \forall i \in [1, n] : \\
& A_i \to \sigma_i \in \mathscr{R} \wedge \check{\delta}' = \varsigma_1 \check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1] \varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n] \varsigma_{n+1} \wedge \check{\tau}' = \alpha^{\check{s}}(\check{\delta}') \\
& \qquad \wr \text{by def. } \alpha^{\check{s}} \text{ and by defining } \check{\delta}' = \varsigma_1 \check{\mathsf{R}}^{\check{\delta}}[A_1 \to \sigma_1] \varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{\delta}}[A_n \to \sigma_n] \varsigma_{n+1} \wr \\
\Longleftrightarrow \quad & \exists \check{\delta} \in \phi(A) : \exists \check{\delta}' : \check{\delta} \ \boxed{\delta}\!\!\Longrightarrow_{\mathscr{G}} \check{\delta}' \wedge \check{\tau}' = \alpha^{\check{s}}(\check{\delta}') \qquad \wr \text{def. (42) of } \boxed{\delta}\!\!\Longrightarrow_{\mathscr{G}} \wr. \quad \square
\end{aligned}
$$

As a corollary of this proof, we have shown that

Corollary 55.

$$\{\alpha^{\check{s}}(\check{\delta}) \mid \exists A \in \mathscr{N} : \boxed{A} \ \boxed{\delta}\!\!\Longrightarrow_{\mathscr{G}} \check{\delta}\} \quad = \quad \{\check{\tau} \mid \exists A \in \mathscr{N} : \boxed{A} \ \boxed{s}\!\!\Longrightarrow_{\mathscr{G}} \check{\tau}\}. \quad \square$$

Corollary 56.

$$\mathsf{S}^{\check{s}}[\![\mathscr{G}]\!] \quad = \quad \boldsymbol{\lambda} A \bullet \{\check{\tau} \in \check{\mathscr{T}} \mid \boxed{A} \ \boxed{s}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}} \check{\tau}\}. \quad \square$$

Proof. By Theorem 54, $\mathsf{S}^{\check{s}}[\![\mathscr{G}]\!] = \mathbf{lfp}^{\subseteq}\ \check{\mathsf{F}}^{\check{s}}[\![\mathscr{G}]\!]$ where $\check{\mathsf{F}}^{\check{s}}[\![\mathscr{G}]\!] = \boldsymbol{\lambda} \phi \bullet \boldsymbol{\lambda} A \bullet \{\boxed{A}\} \cup \mathrm{post}[\boxed{s}\!\!\Longrightarrow_{\mathscr{G}}]\phi(A)$ so $\mathsf{S}^{\check{s}}[\![\mathscr{G}]\!](A) = \mathbf{lfp}^{\subseteq}\ \boldsymbol{\lambda} X \bullet \{\boxed{A}\} \cup \mathrm{post}[\boxed{s}\!\!\Longrightarrow_{\mathscr{G}}]X$ by Example 105 whence $\mathsf{S}^{\check{s}}[\![\mathscr{G}]\!](A) = \mathrm{post}[\boxed{s}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}}](\{\boxed{A}\}) = \{\check{\tau} \in \check{\mathscr{T}} \mid \boxed{A} \ \boxed{s}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}} \check{\tau}\}$ by (A.1). $\square$

17.3. Fixpoint top-down protolanguage semantics

17.3.1. Protolanguage semantics

The *protolanguage semantics* $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] \in \mathscr{N} \mapsto \wp(\mathscr{V}^\star)$ of a context-free grammar $\mathscr{G} = \langle \mathscr{T}, \mathscr{N}, \overline{S}, \mathscr{R} \rangle$ is the protolanguage generated by the grammar $\mathscr{G}$ for each nonterminal. It is defined as

$$\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] \quad \triangleq \quad \alpha^{\check{L}}(\mathsf{S}^{\check{s}}[\![\mathscr{G}]\!]). \tag{46}$$

17.3.2. Protolanguage derivation

Let us define the *protolanguage derivation* $\Longrightarrow_{\mathscr{G}}$ for a grammar $\mathscr{G} = \langle \mathscr{T}, \mathscr{N}, \overline{S}, \mathscr{R} \rangle$ ($\Longrightarrow$ when $\mathscr{G}$ is understood)

$$
\begin{aligned}
& \eta \Longrightarrow_{\mathscr{G}} \eta' \qquad (47) \\
\triangleq \quad & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n, \sigma_1, \ldots, \sigma_n : \eta = \varsigma_1 A_1 \varsigma_2 \ldots \varsigma_n A_n \varsigma_{n+1} \wedge \\
& \quad \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \eta' = \varsigma_1 \sigma_1 \varsigma_2 \ldots \varsigma_n \sigma_n \varsigma_{n+1}.
\end{aligned}
$$

This is [8, Def. 8.2.2] for $n = 1$, the difference being that we allow several simultaneous substitutions.

17.3.3. Fixpoint top-down structural protolanguage semantics

The protolanguage semantics can be defined in fixpoint form as

Theorem 57.

$$\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] \quad = \quad \textit{\textbf{lfp}}^{\dot{\subseteq}}\ \check{\mathsf{F}}^{\check{L}}[\![\mathscr{G}]\!] \quad \textit{where} \quad \check{\mathsf{F}}^{\check{L}}[\![\mathscr{G}]\!] \quad \stackrel{\Delta}{=} \quad \boldsymbol{\lambda}\,\phi \bullet \boldsymbol{\lambda}\,A \bullet \{A\} \cup \mathrm{post}[\Longmapsto_{\mathscr{G}}]\phi(A). \quad \square$$

Proof. We apply Theorem 49 to the fixpoint characterization Theorem 54 of $\mathsf{S}^{\check{s}}[\![\mathscr{G}]\!] = \textbf{lfp}^{\dot{\subseteq}}\ \check{\check{\mathsf{F}}}^{\check{s}}[\![\mathscr{G}]\!]$. We have $\alpha^{\check{L}}(\boldsymbol{\lambda}\,A \bullet \{\boxed{A}\}) = \boldsymbol{\lambda}\,A \bullet \{A\}$ and given an iterate ϕ of $\check{\check{\mathsf{F}}}^{\check{s}}[\![\mathscr{G}]\!]$, we have

$$
\begin{aligned}
& \alpha^{\check{L}}(\boldsymbol{\lambda}\,A \bullet \mathrm{post}[\boxed{s}\!\Longmapsto_{\mathscr{G}}]\phi(A)) \\
= \quad & \boldsymbol{\lambda}\,A \bullet \{\alpha^{\check{L}}(\check{\tau}') \mid \exists \check{\tau} \in \phi(A) : \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}'\} && \wr\text{def. } \alpha^{\check{L}} \text{ and post}\wr \\
= \quad & \boldsymbol{\lambda}\,A \bullet \{\eta' \mid \exists \eta \in \alpha^{\check{L}}(\phi(A)) : \eta \Longmapsto_{\mathscr{G}} \eta'\} \\
& \quad \wr\text{provided we can define } \Longmapsto_{\mathscr{G}} \text{ such that } \{\alpha^{\check{L}}(\check{\tau}') \mid \exists \check{\tau} \in \phi(A) : \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}'\} = \{\eta' \mid \exists \eta \in \alpha^{\check{L}}(\phi(A)) : \eta \Longmapsto_{\mathscr{G}} \eta'\}\wr \\
= \quad & \boldsymbol{\lambda}\,A \bullet \mathrm{post}[\Longmapsto_{\mathscr{G}}](\alpha^{\check{L}}(\phi)(A)) && \wr\text{def. post and } \alpha^{\check{L}}\wr.
\end{aligned}
$$

The design of $\Longmapsto_{\mathscr{G}}$ derives from the condition

$$
\begin{aligned}
& \{\alpha^{\check{L}}(\check{\tau}') \mid \exists \check{\tau} \in \phi(A) : \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}'\} \subseteq \{\eta' \mid \exists \eta \in \alpha^{\check{L}}(\phi(A)) : \eta \Longmapsto_{\mathscr{G}} \eta'\} \\
\Longleftrightarrow \quad & \forall \check{\tau} \in \phi(A) : \forall \check{\tau}' : (\check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}') \Longrightarrow (\exists \eta \in \alpha^{\check{L}}(\phi(A)) : \eta \Longmapsto_{\mathscr{G}} \alpha^{\check{L}}(\check{\tau}')) && \wr\text{def. } \subseteq, \exists\wr \\
\Longleftarrow \quad & \forall \check{\tau} \in \phi(A) : \forall \check{\tau}' : (\check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}') \Longrightarrow (\alpha^{\check{L}}(\check{\tau}) \Longmapsto_{\mathscr{G}} \alpha^{\check{L}}(\check{\tau}')) && \wr\text{choosing } \eta = \alpha^{\check{L}}(\check{\tau})\wr
\end{aligned}
$$

as follows

$$
\begin{aligned}
& \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}' \\
\Longrightarrow \quad & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{L}}(\check{\tau}) = \alpha^{\check{L}}(\varsigma_1)A_1\alpha^{\check{L}}(\varsigma_2)\ldots\alpha^{\check{L}}(\varsigma_n)A_n\alpha^{\check{L}}(\varsigma_{n+1}) \wedge \forall i \in \\
& [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \alpha^{\check{L}}(\check{\tau}') = \alpha^{\check{L}}(\varsigma_1)\alpha^{\check{L}}(\check{\check{\mathsf{R}}}^{\check{s}}[A_1 \to \sigma_1])\alpha^{\check{L}}(\varsigma_2)\ldots\alpha^{\check{L}}(\varsigma_n)\alpha^{\check{L}}(\check{\check{\mathsf{R}}}^{\check{s}}[A_n \to \sigma_n])\alpha^{\check{L}}(\varsigma_{n+1}) \\
& && \wr\text{def. (45) of } \boxed{s}\!\Longmapsto_{\mathscr{G}}, =, \text{and } \alpha^{\check{L}}\wr \\
\Longleftrightarrow \quad & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{L}}(\check{\tau}) = \alpha^{\check{L}}(\varsigma_1)A_1\alpha^{\check{L}}(\varsigma_2)\ldots\alpha^{\check{L}}(\varsigma_n)A_n\alpha^{\check{L}}(\varsigma_{n+1}) \wedge \forall i \in \\
& [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \alpha^{\check{L}}(\check{\tau}') = \alpha^{\check{L}}(\varsigma_1)\sigma_1\alpha^{\check{L}}(\varsigma_2)\ldots\alpha^{\check{L}}(\varsigma_n)\sigma_n\alpha^{\check{L}}(\varsigma_{n+1}) \\
& && \wr\text{def. } \alpha^{\check{L}} \text{ and (44) of } \check{\mathsf{R}}^{\check{s}} \text{ so that } \alpha^{\check{L}}(\check{\mathsf{R}}^{\check{s}}[A \to \sigma]) = \sigma\wr \\
\Longrightarrow \quad & \exists n > 0, \varsigma'_1, \ldots, \varsigma'_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \alpha^{\check{L}}(\check{\tau}) = \varsigma'_1A_1\varsigma'_2\ldots\varsigma'_nA_n\varsigma'_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \\
& \mathscr{R} \wedge \alpha^{\check{L}}(\check{\tau}') = \varsigma'_1\sigma_1\varsigma'_2\ldots\varsigma'_n\sigma_n - \varsigma_{n+1} && \wr\text{by letting } \varsigma'_i = \alpha^{\check{L}}(\varsigma_i), i = 1, \ldots, n+1\wr \\
\Longleftrightarrow \quad & \alpha^{\check{L}}(\check{\tau}) \Longmapsto_{\mathscr{G}} \alpha^{\check{L}}(\check{\tau}') && \wr\text{by defining } \Longmapsto_{\mathscr{G}} \text{ as in (47)}\wr.
\end{aligned}
$$

Inversely, we must also check that

$$
\begin{aligned}
& \{\eta' \mid \exists \eta \in \alpha^{\check{L}}(\phi(A)) : \eta \Longmapsto_{\mathscr{G}} \eta'\} \subseteq \{\alpha^{\check{L}}(\check{\tau}') \mid \exists \check{\tau} \in \phi(A) : \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}'\} \\
\Longleftrightarrow \quad & \forall \eta \in \alpha^{\check{L}}(\phi(A)) : \forall \eta' : (\eta \Longmapsto_{\mathscr{G}} \eta') \Longrightarrow (\exists \check{\tau} \in \phi(A) : \exists \check{\tau}' : \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}' \wedge \eta' = \alpha^{\check{L}}(\check{\tau}')) && \wr\text{def. } \subseteq\wr \\
\Longleftrightarrow \quad & \forall \check{\tau}'' \in \phi(A) : \forall \eta' : (\alpha^{\check{L}}(\check{\tau}'') \Longmapsto_{\mathscr{G}} \eta') \Longrightarrow (\exists \check{\tau} \in \phi(A) : \exists \check{\tau}' : \check{\tau} \ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}' \wedge \eta' = \alpha^{\check{L}}(\check{\tau}')) \\
& && \wr\text{since } \eta \in \alpha^{\check{L}}(\phi(A)) \text{ so } \eta = \alpha^{\check{L}}(\check{\tau}'') \text{ for some } \check{\tau}'' \in \phi(A)\wr.
\end{aligned}
$$

We have $\check{\tau}'' \in \phi(A)$ and $\phi(A)$ is an iterate of $\check{\check{\mathsf{F}}}^{\check{s}}[\![\mathscr{G}]\!]$ hence $\boxed{\overline{S}}\ \boxed{s}\!\Longmapsto_{\mathscr{G}} \check{\tau}''$ so by def. (45) of $\boxed{s}\!\Longmapsto_{\mathscr{G}}$, $\check{\tau}''$ has necessarily the form $\varsigma'_1\boxed{A'_1}\varsigma'_2\ldots\varsigma'_m\boxed{A'_m}\varsigma'_{m+1}$ where $m \geqslant 0$.

$$
\begin{aligned}
& \alpha^{\check{L}}(\check{\tau}'') \Longmapsto_{\mathscr{G}} \eta' \\
\Longleftrightarrow \quad & \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n, \sigma_1, \ldots, \sigma_n : \alpha^{\check{L}}(\check{\tau}'') = \varsigma_1A_1\varsigma_2\ldots\varsigma_nA_n\varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \eta' = \\
& \varsigma_1\sigma_1\varsigma_2\ldots\varsigma_n\sigma_n\varsigma_{n+1} && \wr\text{def. (47) of } \Longmapsto_{\mathscr{G}}\wr \\
\Longrightarrow \quad & \exists n > 0, \varsigma''_1, \ldots, \varsigma''_{n+1}, A_1, \ldots, A_n, \sigma_1, \ldots, \sigma_n : \check{\tau}'' = \varsigma''_1\boxed{A'_1}\varsigma'_2\ldots\varsigma''_n\boxed{A'_n}\varsigma''_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \eta' = \\
& \alpha^{\check{L}}(\varsigma''_1)\sigma_1\alpha^{\check{L}}(\varsigma''_2)\ldots\alpha^{\check{L}}(\varsigma''_n)\sigma_n\alpha^{\check{L}}(\varsigma''_{n+1}) \\
& \quad \wr\text{since } \check{\tau}'' = \varsigma'_1\boxed{A'_1}\varsigma'_2\ldots\varsigma'_m\boxed{A'_m}\varsigma'_{m+1} \text{ so } \alpha^{\check{L}}(\check{\tau}'') = \alpha^{\check{L}}(\varsigma'_1)A'_1\alpha^{\check{L}}(\varsigma'_2)\ldots\alpha^{\check{L}}(\varsigma'_m)A'_m\alpha^{\check{L}}(\varsigma'_{m+1}) = \varsigma_1A_1\varsigma_2\ldots\varsigma_nA_n\varsigma_{n+1} \\
& \quad\ \text{hence, by def. of } \alpha^{\check{L}}, \check{\tau}'' \text{ has the form } \varsigma''_1\boxed{A'_1}\varsigma'_2\ldots\varsigma''_n\boxed{A'_n}\varsigma''_{n+1} \text{ with } \alpha^{\check{L}}(\varsigma''_i) = \varsigma_i, i = 1, \ldots, n+1\wr \\
\Longleftrightarrow \quad & \exists n > 0, \varsigma''_1, \ldots, \varsigma''_{n+1}, A_1, \ldots, A_n, \sigma_1, \ldots, \sigma_n : \check{\tau}'' = \varsigma''_1\boxed{A'_1}\varsigma''_2\ldots\varsigma''_n\boxed{A'_n}\varsigma''_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \eta' = \\
& \alpha^{\check{L}}(\varsigma_1)\alpha^{\check{L}}(\check{\check{\mathsf{R}}}^{\check{s}}[A_1 \to \sigma_1])\alpha^{\check{L}}(\varsigma_2)\ldots\alpha^{\check{L}}(\varsigma_n)\alpha^{\check{L}}(\check{\check{\mathsf{R}}}^{\check{s}}[A_n \to \sigma_n])\alpha^{\check{L}}(\varsigma_{n+1}) && \wr\text{def. } \alpha^{\check{L}} \text{ and (44) of } \check{\mathsf{R}}^{\check{s}} \text{ so that} \\
& \alpha^{\check{L}}(\check{\check{\mathsf{R}}}^{\check{s}}[A \to \sigma]) = \sigma\wr
\end{aligned}
$$

$\Longrightarrow\ \exists \check{\tau} \in \phi(A) : \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n, \sigma_1, \ldots, \sigma_n : \check{\tau} = \varsigma_1 \boxed{A'_1} \varsigma_2 \ldots \varsigma_n \boxed{A'_n} \varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \eta' = \alpha^{\check{L}}(\varsigma_1 \check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1]\varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n]\varsigma_{n+1})$

$\wr$def. $\alpha^{\check{L}}$, renaming ς''_i as ς_i, $i = 1, \ldots, n+1$ and choosing $\check{\tau} = \check{\tau}''\wr$

$\Longleftrightarrow\ \exists \check{\tau} \in \phi(A) : \exists \check{\tau}' : \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n \in \mathscr{N}, \sigma_1, \ldots, \sigma_n \in \mathscr{V}^\star : \check{\tau} = \varsigma_1 \boxed{A_1} \varsigma_2 \ldots \varsigma_n \boxed{A_n} \varsigma_{n+1} \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \check{\tau}' = \varsigma_1 \check{\mathsf{R}}^{\check{s}}[A_1 \to \sigma_1]\varsigma_2 \ldots \varsigma_n \check{\mathsf{R}}^{\check{s}}[A_n \to \sigma_n]\varsigma_{n+1} \wedge \eta' = \alpha^{\check{L}}(\check{\tau}')$ $\wr$def. $\exists\wr$

$\Longleftrightarrow\ \exists \check{\tau} \in \phi(A) : \exists \check{\tau}' : \check{\tau} \ \boxed{s}\!\!\Longrightarrow_{\mathscr{G}} \check{\tau}' \wedge \eta' = \alpha^{\check{L}}(\check{\tau}')$ $\wr$def. (45) of $\boxed{s}\!\!\Longrightarrow_{\mathscr{G}}\wr$. $\square$

As a corollary of this proof and (16), it follows that

Corollary 58.

$$\boldsymbol{\lambda} A \bullet \{\alpha^{\check{L}}(\alpha^{\check{s}}(\alpha^{\check{\delta}}(\pi))) \mid (\vdash \xrightarrow{\boxed{A}} \dashv) \ \boxed{D}\!\!\overset{\star}{\Longrightarrow}_{\mathscr{G}} \pi\} \quad = \quad \boldsymbol{\lambda} A \bullet \{\eta \mid A \overset{\star}{\longmapsto}_{\mathscr{G}} \eta\} \qquad \square$$

so that we also have the classical definition of the protolanguage generated by a grammar [8, Def. 8.2.3].

Corollary 59.

$$\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] \quad = \quad \boldsymbol{\lambda} A \bullet \{\eta \in \mathscr{V}^\star \mid A \overset{\star}{\longmapsto}_{\mathscr{G}} \eta\}. \quad \square$$

Proof. By Theorem 57, $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] = \mathbf{lfp}^{\subseteq}\ \check{\mathsf{F}}^{\check{L}}[\![\mathscr{G}]\!]$ where $\check{\mathsf{F}}^{\check{L}}[\![\mathscr{G}]\!] = \boldsymbol{\lambda}\,\phi \bullet \boldsymbol{\lambda} A \bullet \{A\} \cup \mathrm{post}[\longmapsto_{\mathscr{G}}]\phi(A)$ so $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!](A) = \mathbf{lfp}^{\subseteq}\ \boldsymbol{\lambda} X \bullet \{A\} \cup \mathrm{post}[\longmapsto_{\mathscr{G}}]X$ by Example 105 whence $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!](A) = \mathrm{post}[\overset{\star}{\longmapsto}_{\mathscr{G}}](\{A\}) = \{\eta \in \mathscr{V}^\star \mid A \overset{\star}{\longmapsto}_{\mathscr{G}} \eta\}$ by (A.1). $\square$

17.4. Extension of the fixpoint top-down structural protolanguage semantics to grammar rule states

The protolanguage semantics $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] \in \mathscr{N} \mapsto \wp(\mathscr{V}^\star)$ can be extended to grammar rule states $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!] \in \mathscr{R}^{\cdot} \mapsto \wp(\mathscr{V}^\star)$ as follows

$$\begin{aligned}
\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma . a\sigma'] &\triangleq a\,\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma a . \sigma'] \\
\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma . B\sigma'] &\triangleq \mathsf{S}^{\check{L}}[\![\mathscr{G}]\!](B)\,\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma B . \sigma'] \\
\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma .] &\triangleq \epsilon
\end{aligned} \tag{48}$$

so that

Corollary 60.

$$\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma . \sigma'] \quad = \quad \{\varsigma \in \mathscr{V}^\star \mid \sigma' \overset{\star}{\longmapsto}_{\mathscr{G}} \varsigma\}. \quad \square$$

Proof. By induction on the length $|\sigma'|$ of σ'.

— $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma . a\sigma'] \ = \ a\,\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma a . \sigma']$ $\wr$def. (48) of $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!]\wr$

$= a\,\{\varsigma \in \mathscr{V}^\star \mid \sigma' \overset{\star}{\longmapsto}_{\mathscr{G}} \varsigma\}$ $\wr$ind. hyp.$\wr$

$= \{\varsigma' \in \mathscr{V}^\star \mid a\,\sigma' \overset{\star}{\longmapsto}_{\mathscr{G}} \varsigma'\}$ $\wr$def. concatenation, $\overset{\star}{\longmapsto}_{\mathscr{G}}$, $\longmapsto_{\mathscr{G}}$, and letting $\varsigma' = a\,\varsigma\wr$

— $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma . B\sigma'] \ = \ \mathsf{S}^{\check{L}}[\![\mathscr{G}]\!](B)\,\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma B . \sigma']$ $\wr$def. (48) of $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!]\wr$

$= \{\eta \in \mathscr{V}^\star \mid B \overset{\star}{\longmapsto}_{\mathscr{G}} \eta\}\,\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma B . \sigma']$ $\wr$Corollary 59$\wr$

$= \{\eta \in \mathscr{V}^\star \mid B \overset{\star}{\longmapsto}_{\mathscr{G}} \eta\}\,\{\varsigma \in \mathscr{V}^\star \mid \sigma' \overset{\star}{\longmapsto}_{\mathscr{G}} \varsigma\}$ $\wr$ind. hyp.$\wr$

$= \{\varsigma' \in \mathscr{V}^\star \mid B\sigma' \overset{\star}{\longmapsto}_{\mathscr{G}} \varsigma'\}$ $\wr$def. concatenation, $\overset{\star}{\longmapsto}_{\mathscr{G}}$, $\longmapsto_{\mathscr{G}}$, and letting $\varsigma' = \eta\,\varsigma\wr$

— $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!][A \to \sigma .] \ = \ \epsilon$ $\wr$def. $\vec{\mathsf{S}}^{\check{L}}[\![\mathscr{G}]\!]\wr$

$= \{\varsigma \in \mathscr{V}^\star \mid \epsilon \overset{\star}{\longmapsto}_{\mathscr{G}} \varsigma\}$ $\wr$def. $\overset{\star}{\longmapsto}_{\mathscr{G}}$ and $\longmapsto_{\mathscr{G}}\wr$. $\square$

18. Abstraction of top-down grammar semantics into bottom-up semantics

In Sections 11 and 17, we have constructed a hierarchy of top-down semantics while in Sections 8 and 15, we have constructed a hierarchy of bottom-up semantics, as illustrated in Fig. 1.

Top-down semantics compute grammatical structures with nonterminal variables $\boxed{A}$ replacing these nonterminal variables by a function of the right-hand side of corresponding grammar rules $A \to \sigma$. When no nonterminal variable is left in the structure, we get a grammatical information which can also be computed bottom-up. As shown in Section 12 by Theorem 22 in the particular case of protoderivations, bottom-up semantics can, up to an isomorphic projection, be understood as abstractions of top-down ones by restriction to terminal structures, that is, without any nonterminal variable.

As shown in Fig. 1, this can be extended to the hierarchy of semantics, up to an isomorphic projection, as follows.

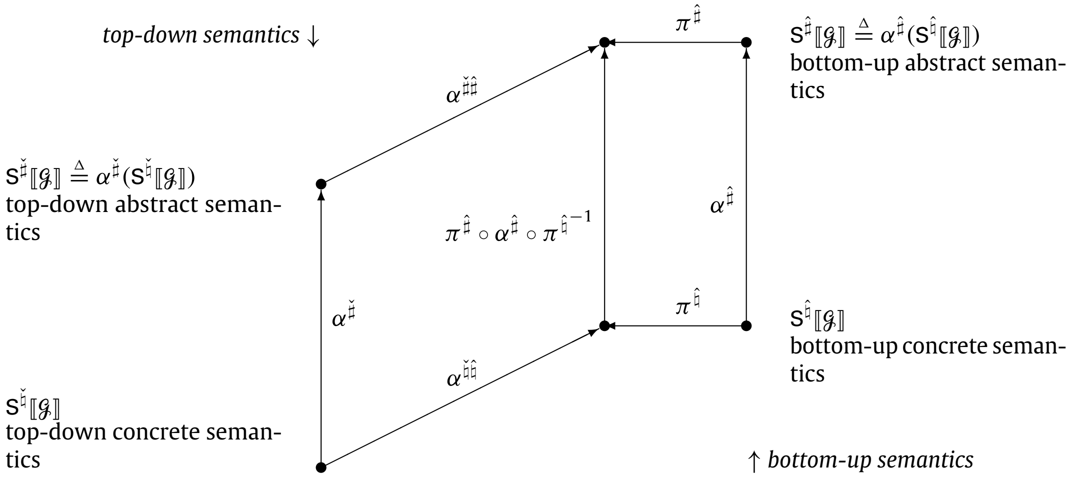


Fig. 6. Top-down to bottom-up abstraction.

Top-down concrete grammar semantics	Abstraction	Bottom-up abstract grammar semantics	Isomorphic projection
Protoderivation $\mathsf{S}^{\check{D}}\llbracket\mathscr{G}\rrbracket$	$\alpha^{\check{D}\hat{d}}$	Derivation $\mathsf{S}^{\hat{d}}\llbracket\mathscr{G}\rrbracket$	$\pi^{\hat{d}} \triangleq \boldsymbol{\lambda}\, T \bullet \boldsymbol{\lambda}\, A \bullet T.A$
Protoderiv. tree $\mathsf{S}^{\check{\delta}}\llbracket\mathscr{G}\rrbracket$	$\alpha^{\check{\delta}\hat{\delta}}$	Derivation tree $\mathsf{S}^{\hat{\delta}}\llbracket\mathscr{G}\rrbracket$	$\pi^{\hat{\delta}} \triangleq \boldsymbol{\lambda}\, T \bullet \boldsymbol{\lambda}\, A \bullet T.A$
Protosyntax tree $\mathsf{S}^{\check{s}}\llbracket\mathscr{G}\rrbracket$	$\alpha^{\check{s}\hat{s}}$	Syntax tree $\mathsf{S}^{\hat{s}}\llbracket\mathscr{G}\rrbracket$	$\pi^{\hat{s}} \triangleq \boldsymbol{\lambda}\, T \bullet \boldsymbol{\lambda}\, A \bullet T.A$
Protolanguage $\mathsf{S}^{\check{L}}\llbracket\mathscr{G}\rrbracket$	$=$	Protolanguage $\mathsf{S}^{\hat{L}}\llbracket\mathscr{G}\rrbracket$	$\pi^{\hat{L}} \triangleq \mathbb{1}$
Protolanguage $\mathsf{S}^{\check{L}}\llbracket\mathscr{G}\rrbracket$	$\dot{\alpha}^{\ell}$	Language $\mathsf{S}^{\ell}\llbracket\mathscr{G}\rrbracket$	$\pi^{\ell} \triangleq \mathbb{1}$

This shows that although the top-down grammar semantics and bottom-up grammar semantics differ in the way derivations, derivation trees and syntax trees are built, they do coincide for protolanguages whence for terminal languages and therefore define the same language, although in different ways.

One level of abstraction in Fig. 1 (where the isomorphic projections are omitted for simplicity) can be described as shown in Fig. 6.

Lemma 61. *If $\pi^{\hat{\natural}}$ is a bijection, $\mathsf{S}^{\check{\sharp}}\llbracket\mathscr{G}\rrbracket \triangleq \alpha^{\check{\sharp}}(\mathsf{S}^{\check{\natural}}\llbracket\mathscr{G}\rrbracket)$, $\mathsf{S}^{\hat{\sharp}}\llbracket\mathscr{G}\rrbracket \triangleq \alpha^{\hat{\sharp}}(\mathsf{S}^{\hat{\natural}}\llbracket\mathscr{G}\rrbracket)$, $\alpha^{\check{\sharp}\hat{\sharp}} \circ \alpha^{\check{\sharp}} = \pi^{\hat{\sharp}} \circ \alpha^{\hat{\sharp}} \circ \pi^{\hat{\natural}^{-1}} \circ \alpha^{\check{\natural}\hat{\natural}}$, $\alpha^{\check{\natural}\hat{\natural}}(\mathsf{S}^{\check{\natural}}\llbracket\mathscr{G}\rrbracket) = \pi^{\hat{\natural}}(\mathsf{S}^{\hat{\natural}}\llbracket\mathscr{G}\rrbracket)$, then $\alpha^{\check{\sharp}\hat{\sharp}}(\mathsf{S}^{\check{\sharp}}\llbracket\mathscr{G}\rrbracket) = \pi^{\hat{\sharp}}(\mathsf{S}^{\hat{\sharp}}\llbracket\mathscr{G}\rrbracket)$.* □

Proof.

$$
\begin{array}{rlr}
& \alpha^{\check{\sharp}\hat{\sharp}}(\mathsf{S}^{\check{\sharp}}\llbracket\mathscr{G}\rrbracket) & \\
= & \alpha^{\check{\sharp}\hat{\sharp}} \circ \alpha^{\check{\sharp}}(\mathsf{S}^{\check{\natural}}\llbracket\mathscr{G}\rrbracket) & \wr\text{since } \mathsf{S}^{\check{\sharp}}\llbracket\mathscr{G}\rrbracket \triangleq \alpha^{\check{\sharp}}(\mathsf{S}^{\check{\natural}}\llbracket\mathscr{G}\rrbracket) \text{ and def. } \circ\wr \\
= & \pi^{\hat{\sharp}} \circ \alpha^{\hat{\sharp}} \circ \pi^{\hat{\natural}^{-1}} \circ \alpha^{\check{\natural}\hat{\natural}}(\mathsf{S}^{\check{\natural}}\llbracket\mathscr{G}\rrbracket) & \wr\text{since } \alpha^{\check{\sharp}\hat{\sharp}} \circ \alpha^{\check{\sharp}} = \pi^{\hat{\sharp}} \circ \alpha^{\hat{\sharp}} \circ \pi^{\hat{\natural}^{-1}} \circ \alpha^{\check{\natural}\hat{\natural}}\wr \\
= & \pi^{\hat{\sharp}} \circ \alpha^{\hat{\sharp}} \circ \pi^{\hat{\natural}^{-1}} \circ \pi^{\hat{\natural}}(\mathsf{S}^{\hat{\natural}}\llbracket\mathscr{G}\rrbracket) & \wr\text{since } \alpha^{\check{\natural}\hat{\natural}}(\mathsf{S}^{\check{\natural}}\llbracket\mathscr{G}\rrbracket) = \pi^{\hat{\natural}}(\mathsf{S}^{\hat{\natural}}\llbracket\mathscr{G}\rrbracket)\wr \\
= & \pi^{\hat{\sharp}}(\mathsf{S}^{\hat{\sharp}}\llbracket\mathscr{G}\rrbracket) & \wr\text{since } \pi^{\hat{\natural}} \text{ is a bijection and } \mathsf{S}^{\hat{\sharp}}\llbracket\mathscr{G}\rrbracket \triangleq \alpha^{\hat{\sharp}}(\mathsf{S}^{\hat{\natural}}\llbracket\mathscr{G}\rrbracket)\wr. \quad \square
\end{array}
$$

18.1. Abstraction of the top-down protoderivation tree grammar semantics into the bottom-up derivation tree semantics

Let us define the abstraction $\alpha^{\check{\delta}\hat{\delta}} \triangleq \boldsymbol{\lambda}\, T \bullet \boldsymbol{\lambda}\, A \bullet T(A) \cap \hat{\mathscr{D}}$ such that

$$\langle \mathscr{N} \mapsto \wp(\check{\mathscr{D}}),\ \dot{\subseteq}\rangle \underset{\alpha^{\check{\delta}\hat{\delta}}}{\overset{\gamma^{\check{\delta}\hat{\delta}}}{\longleftarrow\!\!\!\twoheadrightarrow}} \langle \mathscr{N} \mapsto \wp(\hat{\mathscr{D}}),\ \dot{\subseteq}\rangle$$

which collects the terminal derivation trees (without nonterminal variables) among protoderivation trees.

Lemma 62.

$$\alpha^{\check{\delta}\hat{\delta}} \circ \alpha^{\check{\delta}} \quad = \quad \boldsymbol{\lambda}\, P \in \mathscr{N} \mapsto \wp(\Pi) \bullet \boldsymbol{\lambda}\, A \bullet \alpha^{\hat{\delta}}(\alpha^{\check{D}\hat{d}}(P)A). \quad \square$$

Proof. Given $P \in \mathcal{N} \mapsto \wp(\Pi)$, we calculate

$$\begin{aligned}
&= \alpha^{\check{\delta}\hat{\delta}}(\alpha^{\check{\delta}}(P)) \quad = \quad \boldsymbol{\lambda} A \bullet \alpha^{\check{\delta}}(P(A)) \cap \hat{\mathscr{D}} && \wr\text{def. } \alpha^{\check{\delta}\hat{\delta}} \text{ and } \alpha^{\check{\delta}}\int \\
&= \boldsymbol{\lambda} A \bullet \{\alpha^{\check{\delta}}(\pi) \mid \pi \in P(A)\} \cap \hat{\mathscr{D}} && \wr\text{def. } \alpha^{\check{\delta}} \in \Pi \mapsto \check{\mathscr{D}} \text{ where } \check{\mathscr{D}} \triangleq (\mathscr{P} \cup \check{\mathscr{U}})^\star \text{ and } \check{\mathscr{U}} \triangleq \mathscr{T} \cup \mathcal{N}^\Box \cup \mathscr{R}^\centerdot\int \\
&= \boldsymbol{\lambda} A \bullet \{\alpha^{\hat{\delta}}(\theta) \mid \theta \in (P(A) \cap \Theta)\}
\end{aligned}$$

$\wr$where $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathscr{D}}$, $\hat{\mathscr{D}} \triangleq (\mathscr{P} \cup \hat{\mathscr{U}})^\star$ and $\hat{\mathscr{U}} \triangleq \mathscr{T} \cup \mathscr{R}^\centerdot$ since $\alpha^{\check{\delta}}(\pi) \in \hat{\mathscr{D}}$ if and only if π has not nonterminal variable in $\mathcal{N}^\Box$ that is $\pi \in \Theta\int$

$$\begin{aligned}
&= \boldsymbol{\lambda} A \bullet \alpha^{\hat{\delta}}(P(A) \cap \Theta) && \wr\text{def. } \alpha^{\hat{\delta}} \text{ where } (P(A) \cap \Theta) \in \wp(\Theta)\int \\
&= \boldsymbol{\lambda} A \bullet \alpha^{\hat{\delta}}(\alpha^{\check{D}\hat{d}}(P)A) && \wr\text{def. } \alpha^{\check{D}\hat{d}}\int. \quad \Box
\end{aligned}$$

The protoderivation tree semantics is a top-down way of defining the derivation tree semantics, by restriction to terminal trees, as follows

Theorem 63.

$$\alpha^{\check{\delta}\hat{\delta}}(\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!]) \quad = \quad \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!].A \quad = \quad \boldsymbol{\lambda} A \bullet \{\hat{\delta} \in \hat{\mathscr{D}} \mid \boxed{A}\ \boxed{\check{\delta}}\!\!\Longrightarrow_{\mathcal{G}} \hat{\delta}\}. \quad \Box$$

Proof. As shown in Lemma 61, we have:

$$\begin{aligned}
&\alpha^{\check{\delta}\hat{\delta}}(\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!]) \quad = \quad \alpha^{\check{\delta}\hat{\delta}}(\alpha^{\check{\delta}}(\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!])) && \wr\text{def. (37) of } \mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!]\int \\
&= \boldsymbol{\lambda} A \bullet \alpha^{\hat{\delta}}(\alpha^{\check{D}\hat{d}}(\mathsf{S}^{\check{D}}[\![\mathcal{G}]\!])A) \quad = \quad \boldsymbol{\lambda} A \bullet \alpha^{\hat{\delta}}(\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].A) && \wr\text{by Lemmas 22 and 62}\int \\
&= \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!].A && \wr\text{def. } \alpha^{\hat{\delta}} \text{ and (26) of } \mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!]\int.
\end{aligned}$$

Moreover

$$\begin{aligned}
&\alpha^{\check{\delta}\hat{\delta}}(\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!]) \ = \ \boldsymbol{\lambda} A \bullet \{\alpha^{\check{\delta}}(\pi) \mid \pi \in \mathsf{S}^{\check{D}}[\![\mathcal{G}]\!](A)\} \cap \hat{\mathscr{D}} && \wr\text{def. } \alpha^{\check{\delta}\hat{\delta}}, \text{(37) of } \mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!], \text{ and } \alpha^{\check{\delta}}\int \\
&= \boldsymbol{\lambda} A \bullet \{\alpha^{\check{\delta}}(\pi) \mid \pi \in \Pi \wedge \vdash \xrightarrow{\boxed{A}} \dashv)\ \boxed{\check{D}}\!\!\overset{\star}{\Longrightarrow}_{\mathcal{G}} \pi\} \cap \hat{\mathscr{D}} && \wr\text{def. (16) of } \mathsf{S}^{\check{D}}[\![\mathcal{G}]\!]\int \\
&= \boldsymbol{\lambda} A \bullet \{\alpha^{\check{\delta}}(\pi) \mid \exists A \in \mathcal{N} : \pi \in \Pi \wedge \vdash \xrightarrow{\boxed{A}} \dashv)\ \boxed{\check{D}}\!\!\overset{\star}{\Longrightarrow}_{\mathcal{G}} \pi\}.A \cap \hat{\mathscr{D}} && \wr\text{def. selection } \bullet.\bullet\int \\
&= \boldsymbol{\lambda} A \bullet \{\check{\delta} \mid \exists A \in \mathcal{N} : \boxed{A}\ \boxed{\check{\delta}}\!\!\Longrightarrow_{\mathcal{G}} \check{\delta}\}.A \cap \hat{\mathscr{D}} && \wr\text{by Corollary 52}\int \\
&= \boldsymbol{\lambda} A \bullet \{\hat{\delta} \in \hat{\mathscr{D}} \mid \boxed{A}\ \boxed{\check{\delta}}\!\!\Longrightarrow_{\mathcal{G}} \hat{\delta}\} && \wr\text{def. selection } \bullet.\bullet \text{ and } \cap\int. \quad \Box
\end{aligned}$$

18.2. Abstraction of the top-down protosyntax tree grammar semantics into the bottom-up syntax tree semantics

Let us define the abstraction $\alpha^{\check{s}\hat{s}} \triangleq \boldsymbol{\lambda} T \bullet \boldsymbol{\lambda} A \bullet T(A) \cap \hat{\mathscr{T}}$ such that

$$\langle \mathcal{N} \mapsto \wp(\check{\mathscr{T}}),\ \dot{\subseteq}\rangle \underset{\alpha^{\check{s}\hat{s}}}{\overset{\gamma^{\check{s}\hat{s}}}{\leftrightarrows}} \langle \mathcal{N} \mapsto \wp(\hat{\mathscr{T}}),\ \dot{\subseteq}\rangle$$

which collects the terminal syntax trees (without nonterminal variables) among protosyntax trees.

Lemma 64.

$$\alpha^{\check{s}\hat{s}} \circ \alpha^{\check{s}} \quad = \quad \boldsymbol{\lambda} T \in \mathcal{N} \mapsto \wp(\check{\mathscr{T}}) \bullet \boldsymbol{\lambda} A \bullet \alpha^{\hat{s}}(\alpha^{\check{\delta}\hat{\delta}}(T)A). \quad \Box$$

Proof. Given $T \in \mathcal{N} \mapsto \wp(\check{\mathscr{T}})$, we calculate

$$\begin{aligned}
&= \alpha^{\check{s}\hat{s}}(\alpha^{\check{s}}(T)) \quad = \quad \boldsymbol{\lambda} A \bullet \alpha^{\check{s}}(T(A)) \cap \hat{\mathscr{T}} && \wr\text{def. } \alpha^{\check{s}\hat{s}} \text{ and } \alpha^{\check{s}}\int \\
&= \boldsymbol{\lambda} A \bullet \{\alpha^{\check{s}}(\check{\delta}) \mid \check{\delta} \in T(A)\} \cap \hat{\mathscr{T}}
\end{aligned}$$

$\wr$def. $\alpha^{\check{s}}$, where $\alpha^{\check{s}} \in \check{\mathscr{D}} \mapsto \check{\mathscr{T}}$, $\check{\mathscr{D}} \triangleq (\mathscr{P} \cup \check{\mathscr{U}})^\star$, $\check{\mathscr{U}} \triangleq \mathscr{T} \cup \mathcal{N}^\Box \cup \mathscr{R}^\centerdot$, $\check{\mathscr{T}} \triangleq (\mathscr{P} \cup \mathscr{T} \cup \mathcal{N}^\Box)^\star$ and $\hat{\mathscr{T}} \triangleq (\mathscr{P} \cup \mathscr{T})^\star\int$

$$= \boldsymbol{\lambda} A \bullet \{\alpha^{\check{s}}(\check{\delta}) \mid \check{\delta} \in T(A) \cap \hat{\mathscr{D}}\}$$

$\wr$by def. $\alpha^{\check{s}}$ since $\alpha^{\check{s}}(\check{\delta}) \in \hat{\mathscr{T}}$ if and only if $\check{\delta}$ contains no nonterminal variable in $\mathcal{N}^\Box$ that is $\check{\delta} \in \hat{\mathscr{D}}$ where $\hat{\mathscr{D}} \triangleq (\mathscr{P} \cup \hat{\mathscr{U}})^\star$ and $\hat{\mathscr{U}} \triangleq \mathscr{T} \cup \mathscr{R}^\centerdot\int$

$$\begin{aligned}
&= \boldsymbol{\lambda} A \bullet \{\alpha^{\hat{s}}(\hat{\delta}) \mid \hat{\delta} \in (T(A) \cap \hat{\mathscr{D}})\} && \wr\text{by def. } \alpha^{\check{s}} \text{ and } \alpha^{\hat{s}} \text{ which coincide on } \hat{\mathscr{D}}\int \\
&= \boldsymbol{\lambda} A \bullet \alpha^{\hat{s}}(\alpha^{\check{\delta}\hat{\delta}}(T)A) && \wr\text{def. } \alpha^{\hat{s}} \text{ and } \alpha^{\check{\delta}\hat{\delta}}\int. \quad \Box
\end{aligned}$$

The protosyntax tree semantics is a top-down way of defining the syntax tree semantics, by restriction to terminal syntax trees, as follows

Theorem 65.

$$\alpha^{\check{s}\hat{s}}(\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!]) \quad = \quad \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!].A \quad = \quad \boldsymbol{\lambda} A \bullet \{\hat{\tau} \in \hat{\mathcal{T}} \mid \boxed{A} \overset{\boxed{s}}{\Longrightarrow}_{\mathcal{G}} \hat{\tau}\}. \quad \square$$

Proof. As shown in Lemma 61, we have:

$$\begin{aligned}
& \alpha^{\check{s}\hat{s}}(\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!]) \quad = \quad \alpha^{\check{s}\hat{s}}(\alpha^{\check{s}}(\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!])) && \text{(def. (43) of } \mathsf{S}^{\check{s}}[\![\mathcal{G}]\!]\text{)}\\
= \; & \boldsymbol{\lambda} A \bullet \alpha^{\hat{s}}(\alpha^{\check{\delta}\hat{\delta}}(\mathsf{S}^{\check{\delta}}[\![\mathcal{G}]\!])A) && \text{(by Lemma 64)}\\
= \; & \boldsymbol{\lambda} A \bullet \alpha^{\hat{s}}((\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!]).A) && \text{(by Theorem 63)}\\
= \; & \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!].A && \text{(def. } \alpha^{\hat{s}}\text{, selection } \bullet.\bullet\text{, and (29) of } \mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!]\text{)} .
\end{aligned}$$

Moreover

$$\begin{aligned}
& \boldsymbol{\lambda} A \bullet \mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!].A \quad = \quad \boldsymbol{\lambda} A \bullet \alpha^{\hat{s}}((\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!]).A) && \text{(as shown above)}\\
= \; & \boldsymbol{\lambda} A \bullet \alpha^{\hat{s}}(\{\hat{\delta} \in \hat{\mathcal{D}} \mid \boxed{A} \overset{\boxed{\delta}}{\Longrightarrow}_{\mathcal{G}} \hat{\delta}\}) && \text{(by Theorem 63)}\\
= \; & \boldsymbol{\lambda} A \bullet (\{\alpha^{\hat{s}}(\hat{\delta}) \mid \hat{\delta} \in \hat{\mathcal{D}} \wedge \exists A \in \mathcal{N} : \boxed{A} \overset{\boxed{\delta}}{\Longrightarrow}_{\mathcal{G}} \hat{\delta}\}).A) && \text{(def. selection } \bullet.\bullet \text{ and } \alpha^{\hat{s}}\text{)}\\
= \; & \boldsymbol{\lambda} A \bullet (\{\alpha^{\check{s}}(\check{\delta}) \mid \check{\delta} \in \hat{\mathcal{D}} \wedge \exists A \in \mathcal{N} : \boxed{A} \overset{\boxed{\delta}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}\}).A && \text{(by def. } \alpha^{\check{s}} \text{ and } \alpha^{\hat{s}} \text{ which coincide on } \hat{\mathcal{D}}\text{)}\\
= \; & \boldsymbol{\lambda} A \bullet (\{\alpha^{\check{s}}(\check{\delta}) \mid \exists A \in \mathcal{N} : \boxed{A} \overset{\boxed{\delta}}{\Longrightarrow}_{\mathcal{G}} \check{\delta}\} \cap \hat{\mathcal{T}}).A
\end{aligned}$$

(by def. $\alpha^{\check{s}}$ since $\alpha^{\check{s}}(\check{\delta}) \in \hat{\mathcal{T}}$ if and only if $\check{\delta}$ contains no nonterminal variable in $\mathcal{N}^{\square}$ that is $\check{\delta} \in \hat{\mathcal{D}}$ where $\hat{\mathcal{D}} \triangleq (\mathcal{P} \cup \hat{\mathcal{U}})^{\star}$ and $\hat{\mathcal{U}} \triangleq \mathcal{T} \cup \mathcal{R}^{\cdot}$)

$$\begin{aligned}
= \; & \boldsymbol{\lambda} A \bullet (\{\hat{\tau} \mid \exists A \in \mathcal{N} : \boxed{A} \overset{\boxed{s}}{\Longrightarrow}_{\mathcal{G}} \hat{\tau}\} \cap \hat{\mathcal{T}}).A && \text{(Corollary 55)}\\
= \; & \boldsymbol{\lambda} A \bullet \{\hat{\tau} \in \hat{\mathcal{T}} \mid \boxed{A} \overset{\boxed{s}}{\Longrightarrow}_{\mathcal{G}} \hat{\tau}\} && \text{(def. } \cap \text{ and selection } \bullet.\bullet\text{)}. \quad \square
\end{aligned}$$

18.3. Abstraction of the top-down protolanguage grammar semantics into the bottom-up protolanguage semantics

We consider the abstraction $\alpha^{\mathbb{P}} \in \wp(\mathcal{V}^{\star} \times \mathcal{V}^{\star}) \mapsto \mathcal{N} \mapsto \wp(\mathcal{V}^{\star})$ defined as

$$\alpha^{\mathbb{P}}(r) \quad \triangleq \quad \boldsymbol{\lambda} A \bullet \{\sigma \in \mathcal{V}^{\star} \mid \langle A,\ \sigma \rangle \in r\} \quad = \quad \boldsymbol{\lambda} A \bullet \mathrm{post}[r](\{A\})$$

so that $\langle \wp(\mathcal{V}^{\star} \times \mathcal{V}^{\star}),\ \dot{\subseteq} \rangle \xleftarrow[\alpha^{\mathbb{P}}]{\gamma^{\mathbb{P}}} \langle \mathcal{N} \mapsto \wp(\mathcal{V}^{\star}),\ \dot{\subseteq} \rangle$, pointwise.

Lemma 66. *Let F^n, $n \in \mathbb{N}$ be the iterates of $\hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!]$ from $\varnothing$ (as defined in Appendix A.1) with limit $\boldsymbol{lfp}^{\subseteq}\, \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!] = F^{\omega} = \bigcup_{n \in \mathbb{N}} F^n$. $\alpha^{\mathbb{P}}(\varnothing) = \boldsymbol{\lambda} A \bullet \varnothing = F^0$. For $n > 0$, we have $\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}}) = F^n$.* $\square$

Proof. We first prove that $\forall n \geqslant 0 : \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma.\sigma'](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}})) = \{\varsigma \mid \sigma' \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma\}$ by natural induction on the length $|\sigma'|$ of σ'. We have three cases.

$$\begin{aligned}
— \; & \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma.a\sigma'](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}})) \quad = \quad a\, \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma a.\sigma'](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}})) && \text{(def. } \hat{\mathsf{F}}^{\hat{L}} \text{ \& } \overset{n*}{\Longrightarrow}_{\mathcal{G}}\text{)}\\
= \; & a\, \{\varsigma \mid \sigma' \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma\} && \text{(ind. hyp.)}\\
= \; & \{\varsigma' \mid a\sigma' \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma'\} && \text{(def. concatenation, } \overset{n*}{\Longrightarrow}_{\mathcal{G}}, \Longrightarrow_{\mathcal{G}}, \varsigma' = a\varsigma, \text{ def. } \overset{n*}{\Longrightarrow}_{\mathcal{G}} \text{ \& } \Longrightarrow_{\mathcal{G}}\text{)}\\
— \; & \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma.B\sigma'](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}}))\\
= \; & (\{B\} \cup \{\varsigma \mid B \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma\})\, \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma B.\sigma'](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}})) && \text{(def. } \hat{\mathsf{F}}^{\hat{L}} \text{ and } \alpha^{\mathbb{P}}\text{)}\\
= \; & \{\varsigma \mid B \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma\}\, \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma B.\sigma'](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}})) && \text{(} n > 0 \text{ so } \mathbb{1} \subseteq \overset{n*}{\Longrightarrow}_{\mathcal{G}}\text{)}\\
= \; & \{\varsigma \mid B \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma\}\, \{\varsigma' \mid \sigma' \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma'\} && \text{(ind. hyp.)}\\
= \; & \{\varsigma'' \mid B\sigma' \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma''\} && \text{(def. concatenation, } \varsigma'' = \varsigma\varsigma'\text{, def. } \overset{n*}{\Longrightarrow}_{\mathcal{G}} \text{ \& } \Longrightarrow_{\mathcal{G}}\text{)}\\
— \; & \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to \sigma.](\alpha^{\mathbb{P}}(\overset{n*}{\Longrightarrow}_{\mathcal{G}})) \;=\; \{\epsilon\} \;=\; \{\varsigma \mid \epsilon \overset{n*}{\Longrightarrow}_{\mathcal{G}} \varsigma\} && \text{(def. } \hat{\mathsf{F}}^{\hat{L}}, \overset{1*}{\Longrightarrow}_{\mathcal{G}} = \mathbb{1}, \overset{n*}{\Longrightarrow}_{\mathcal{G}} \text{ \& } \Longrightarrow_{\mathcal{G}}\text{)}
\end{aligned}$$

The proof of the lemma is by recurrence on n. For the base case $n = 1$, we have

$$\begin{aligned}
& \alpha^{\mathbb{P}}(\overset{1*}{\Longrightarrow}_{\mathcal{G}}) \quad = \quad \alpha^{\mathbb{P}}(\mathbb{1}) \cup \alpha^{\mathbb{P}}(\Longrightarrow_{\mathcal{G}}) && \text{(def. } \overset{1*}{\Longrightarrow}_{\mathcal{G}}, \alpha^{\mathbb{P}} \text{ preserves lubs, and def. } \overset{n}{\Longrightarrow}_{\mathcal{G}}\text{)}\\
= \; & \boldsymbol{\lambda} A \bullet \{A\} \cup \bigcup_{A \to \sigma \in \mathcal{R}} \{\sigma\} && \text{(def. } \alpha^{\mathbb{P}} \text{ \& } \Longrightarrow_{\mathcal{G}}\text{)}\\
= \; & \boldsymbol{\lambda} A \bullet \{A\} \cup \bigcup_{A \to \sigma \in \mathcal{R}} \hat{\mathsf{F}}^{\hat{L}}_{.}[A \to .\sigma](\boldsymbol{\lambda} B \bullet \varnothing) && \text{(def. } \hat{\mathsf{F}}^{\hat{L}}\text{)}\\
= \; & \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!](F^0) \quad = \quad F^1 && \text{(def. } \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!]\text{, and iterates } F^0, F^1\text{)}.
\end{aligned}$$

For the induction step $n > 1$, we calculate $\alpha^{\mathbb{P}}(\stackrel{n+1*}{\Longmapsto}_{\mathcal{G}})$

$$= \lambda A \bullet \{A\} \cup \bigcup_{A \to \sigma \in \mathcal{R}} \{\varsigma \mid \sigma \stackrel{n*}{\Longmapsto}_{\mathcal{G}} \varsigma\} \qquad \text{≀def. } \stackrel{n+1*}{\Longmapsto}_{\mathcal{G}}, \alpha^{\mathbb{P}} \text{ preserving lubs, } \circ \,\&\, \Longmapsto_{\mathcal{G}} \text{≀}$$

$$= \lambda A \bullet \{A\} \cup \bigcup_{A \to \sigma \in \mathcal{R}} \hat{\mathsf{F}}^{\hat{L}}_{\cdot}[A \to .\sigma](\alpha^{\mathbb{P}}(\stackrel{n*}{\Longmapsto}_{\mathcal{G}})) \qquad \text{≀as shown above≀}$$

$$= \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!](\alpha^{\mathbb{P}}(\stackrel{n*}{\Longmapsto}_{\mathcal{G}})) = \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!](F^n) = F^{n+1} \qquad \text{≀def. } \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!]\text{, ind. hyp., and iterates≀.} \quad \square$$

The classical characterization of the protolanguage generated by a grammar [8, Def. 8.2.3] is

Theorem 67.

$$\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \;=\; \lambda A \bullet \{\sigma \in \mathcal{V}^\star \mid A \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \sigma\}. \quad \square$$

Proof. We must prove that $\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] = \alpha^{\mathbb{P}}(\stackrel{\star}{\Longmapsto}_{\mathcal{G}})$. We have

$$\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \;=\; F^0 \cup F^1 \cup \bigcup_{n>1} F^n \qquad \text{≀Theorem 41, } \hat{\mathsf{F}}^{\hat{L}}[\![\mathcal{G}]\!] \text{ preserves lubs and def. } \textstyle\bigcup\text{≀}$$

$$= \alpha^{\mathbb{P}}(\stackrel{1*}{\Longmapsto}_{\mathcal{G}}) \cup \bigcup_{n>1} \alpha^{\mathbb{P}}(\stackrel{n*}{\Longmapsto}_{\mathcal{G}}) \qquad \text{≀by Lemma 66≀}$$

$$= \alpha^{\mathbb{P}}\left(\bigcup_{n\geqslant 1} \stackrel{n*}{\Longmapsto}_{\mathcal{G}}\right) \;=\; \alpha^{\mathbb{P}}\Big(\bigcup_{n\geqslant 1}\bigcup_{i\leqslant n} \stackrel{i}{\Longmapsto}_{\mathcal{G}}\Big) \qquad \text{≀}\alpha^{\mathbb{P}} \text{ preserves lubs and def. } \stackrel{n*}{\Longmapsto}_{\mathcal{G}}\text{≀}$$

$$= \alpha^{\mathbb{P}}\left(\bigcup_{n\in\mathbb{N}} \stackrel{n}{\Longmapsto}_{\mathcal{G}}\right) \;=\; \alpha^{\mathbb{P}}(\stackrel{\star}{\Longmapsto}_{\mathcal{G}}) \qquad \text{≀def. } \textstyle\bigcup \text{ and } \stackrel{\star}{\Longmapsto}_{\mathcal{G}}\text{≀.} \quad \square$$

It follows that the bottom-up and top-down protolanguage semantics of a grammar are identical (which was not the case at more concrete levels of abstraction).

Corollary 68.

$$\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \;=\; \mathsf{S}^{\check{L}}[\![\mathcal{G}]\!]. \quad \square$$

Proof. $\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] = \lambda A \bullet \{\sigma \in \mathcal{V}^\star \mid A \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \sigma\} = \mathsf{S}^{\check{L}}[\![\mathcal{G}]\!]$ by Theorem 67 and Corollary 59. $\square$

It follows that

Corollary 69.

$$\lambda A \bullet \alpha^{\hat{L}}(\alpha^{\check{s}\hat{s}}(\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!])A) \;=\; \alpha^{\check{L}}(\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!]). \quad \square$$

Proof.

$$\lambda A \bullet \alpha^{\hat{L}}(\alpha^{\check{s}\hat{s}}(\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!])A) \;=\; \mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!] \qquad \text{≀def. } \dot{\alpha}^{\hat{L}} \text{ and (31) of } \mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!]\text{≀}$$

$$= \mathsf{S}^{\check{L}}[\![\mathcal{G}]\!] \;=\; \alpha^{\check{L}}(\mathsf{S}^{\check{s}}[\![\mathcal{G}]\!]) \qquad \text{≀by Corollary 68 and def. (46) of } \mathsf{S}^{\check{L}}[\![\mathcal{G}]\!]\text{≀.} \quad \square$$

However, in general, we have $\alpha^{\check{L}}(T) \neq \lambda A \bullet \alpha^{\hat{L}}(\alpha^{\check{s}\hat{s}}(T)A)$, as shown by the following counterexample.

Example 70. By the choice of T represented by its graph so that $T(A) = \{(\!|A\ \boxed{A}\ \boxed{A}\ A|\!)\}$, we have

$$\alpha^{\check{L}}(T) \;=\; \alpha^{\check{L}}(\{\langle A,\ \{(\!|A\ \boxed{A}\ \boxed{A}\ A|\!)\}\rangle\}) \;=\; \lambda A \bullet \alpha^{\check{L}}(\{(\!|A\ \boxed{A}\ \boxed{A}\ A|\!)\}) \qquad \text{≀def. } \alpha^{\check{L}}\text{≀}$$

$$= \lambda A \bullet \{\alpha^{\check{L}}((\!|A\ \boxed{A}\ \boxed{A}\ A|\!))\} \qquad \text{≀def. } \alpha^{\check{L}}\text{≀}$$

$$\neq \lambda A \bullet \varnothing \;=\; \lambda A \bullet \alpha^{\hat{L}}(\varnothing) \;=\; \lambda A \bullet \alpha^{\hat{L}}(\{(\!|A\ \boxed{A}\ \boxed{A}\ A|\!)\}) \cap \hat{\mathcal{T}}$$

$$\text{≀def. } \alpha^{\hat{L}} \text{ and } \hat{\mathcal{T}} \triangleq (\mathcal{P} \cup \mathcal{T})^\star \text{ with no terminal variables } \boxed{A} \in \mathcal{N}^\square\text{≀}$$

$$= \lambda A \bullet \alpha^{\hat{L}}(\alpha^{\check{s}\hat{s}}(T)A) \qquad \text{≀def. } \alpha^{\check{s}\hat{s}} \text{ and } T = \{\langle A,\ \{(\!|A\ \boxed{A}\ \boxed{A}\ A|\!)\}\rangle\}\text{≀.} \quad \square$$

18.4. Abstraction of the top-down protolanguage grammar semantics into the bottom-up terminal language semantics

Applying the terminal language abstraction, we get the classical definition of the terminal language generated by a grammar [8, Def. 8.2.3]

Theorem 71.

$$\mathsf{S}^{\ell}[\![\mathcal{G}]\!] \ \stackrel{\Delta}{=}\ \dot{\alpha}^{\ell}(\mathsf{S}^{\hat{\mathcal{L}}}[\![\mathcal{G}]\!]) \ =\ \boldsymbol{\lambda} A \bullet \{\sigma \in \mathcal{T}^{\star} \mid A \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \sigma\}. \quad \square$$

Proof. We calculate

$$\begin{array}{rll} & \mathsf{S}^{\ell}[\![\mathcal{G}]\!] \ =\ \dot{\alpha}^{\ell}(\boldsymbol{\lambda} A \bullet \{\sigma \in \mathcal{V}^{\star} \mid A \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \sigma\}) & \wr \text{def. } \mathsf{S}^{\ell}[\![\mathcal{G}]\!] \text{ and Theorem 67} \wr \\ = & \boldsymbol{\lambda} A \bullet \{\sigma \in \mathcal{T}^{\star} \mid A \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \sigma\} & \wr \text{def. } \dot{\alpha}^{\ell}, \alpha^{\ell}, \text{ and } \mathcal{V}^{\star} \cap \mathcal{T}^{\star} = \mathcal{T}^{\star} \wr. \quad \square \end{array}$$

19. Bottom-up grammar analysis

Classical grammar analysis algorithms such as First [8, Section 8.2.8], nonterminal productivity [8, Section 8.2.4], and ϵ-productivity ϵ-Prod [8, Section 8.2.3] are abstractions of the bottom-up grammar semantics and are instances of the bottom-up abstract interpreter (18).

19.1. First

The *first abstraction* $\alpha^1 \in \mathcal{T}^{\star} \mapsto \wp(\mathcal{T} \cup \{\epsilon\})$ of a terminal sentence is the first terminal of this sentence or ϵ for empty sentences. $\alpha^1 \stackrel{\Delta}{=} \boldsymbol{\lambda} \sigma \bullet \{a \in \mathcal{T} \mid \exists \sigma' \in \mathcal{T}^{\star} : \sigma = a\sigma'\} \cup \{\epsilon \mid \sigma = \epsilon\}$. It is extended to terminal languages $\alpha^1 \in \wp(\mathcal{T}^{\star}) \mapsto \wp(\mathcal{T} \cup \{\epsilon\})$ in order to collect the first terminals of the sentences of these languages $\alpha^1 \stackrel{\Delta}{=} \boldsymbol{\lambda} \Sigma \bullet \bigcup_{\sigma \in \Sigma} \alpha^1(\sigma)$ and finally extended pointwise $\dot{\alpha}^1 \in (\mathcal{N} \mapsto \wp(\mathcal{T}^{\star})) \mapsto (\mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\epsilon\}))$ on terminal languages derived for nonterminals as $\dot{\alpha}^1 \stackrel{\Delta}{=} \boldsymbol{\lambda} L \bullet \boldsymbol{\lambda} A \bullet \alpha^1(L(A))$.

The first abstraction of language concatenation is

Lemma 72. *For all $\Sigma, \Sigma' \in \wp(\mathcal{T}^{\star})$ and $F, F' \in \wp(\mathcal{T})$, $\alpha^1(\Sigma\Sigma') = \alpha^1(\Sigma) \oplus^1 \alpha^1(\Sigma')$*

$$\begin{array}{lrcl} \textit{where} & F \oplus^1 F' & \stackrel{\Delta}{=} & (\!|F' \neq \varnothing \ ?\ (F \setminus \{\epsilon\}) \cup (\!|\epsilon \in F \ ?\ F' \ \text{\textcolon}\ \varnothing|\!) \ \text{\textcolon}\ \varnothing|\!) \\ \textit{and} & \{a\} \oplus^1 F' & \stackrel{\Delta}{=} & (\!|F' \neq \varnothing \ ?\ \{a\} \ \text{\textcolon}\ \varnothing|\!). \quad \square \end{array}$$

Proof. We calculate

$$\begin{array}{rl} - & \alpha^1(\Sigma\Sigma') = \bigcup_{\sigma \in \Sigma\Sigma'} \alpha^1(\sigma) = \bigcup_{\sigma_1 \in \Sigma, \sigma_2 \in \Sigma'} \alpha^1(\sigma_1\sigma_2) \qquad \wr \text{def. } \alpha^1 \text{ \& language concat. } \Sigma\Sigma' \wr \\ = & \{a \in \mathcal{T} \mid \exists \sigma, \sigma_1, \sigma_2 : \sigma_1 \in \Sigma \wedge \sigma_2 \in \Sigma' \wedge \sigma_1\sigma_2 = a\sigma\} \cup \{\epsilon \mid \epsilon \in \Sigma \wedge \epsilon \in \Sigma'\} \\ & \qquad \wr \text{def. } \alpha^1, \cup, \text{\& sentence concatenation } \sigma_1\sigma_2 \wr \\ = & \{a \in \mathcal{T} \mid \exists \sigma_1, \sigma_2 : a\sigma_1 \in \Sigma \wedge \sigma_2 \in \Sigma'\} \cup \{a \in \mathcal{T} \mid \exists \sigma_2 : \epsilon \in \Sigma \wedge a\sigma_2 \in \Sigma'\} \cup \{\epsilon \mid \epsilon \in \Sigma \wedge \epsilon \in \Sigma'\} \ \wr \sigma_1\sigma_2 = a\sigma \text{ with } \\ & \sigma_1 \neq \epsilon \text{ or } \sigma_1 = \epsilon \wr \\ = & (\!|\Sigma' \neq \varnothing \ ?\ ((\{a \in \mathcal{T} \mid \exists \sigma_1 : a\sigma_1 \in \Sigma\} \cup \{\epsilon \mid \epsilon \in \Sigma\}) \setminus \{\epsilon\}) \cup \{a \in \mathcal{T} \mid \exists \sigma_2 : \epsilon \in \Sigma \wedge a\sigma_2 \in \Sigma'\} \cup \{\epsilon \mid \epsilon \in \Sigma \wedge \epsilon \in \\ & \Sigma'\} \ \text{\textcolon}\ \varnothing|\!) \qquad \wr \exists \sigma_2 : \sigma_2 \in \Sigma' \Longleftrightarrow \Sigma' \neq \varnothing \text{ and } \epsilon \notin \mathcal{T} \wr \\ = & (\!|\Sigma' \neq \varnothing \ ?\ \alpha^1(\Sigma) \setminus \{\epsilon\} \cup (\!|\epsilon \in \alpha^1(\Sigma) \ ?\ \alpha^1(\Sigma') \ \text{\textcolon}\ \varnothing|\!) \ \text{\textcolon}\ \varnothing|\!) \\ & \qquad \wr \text{def. } \alpha^1\ (\Sigma), \text{ conditional, and } \alpha^1 \text{ so that } \epsilon \in \Sigma \Longleftrightarrow \epsilon \in \alpha^1(\Sigma) \wr \\ = & (\!|\alpha^1(\Sigma') \neq \varnothing \ ?\ \alpha^1(\Sigma) \setminus \{\epsilon\} \cup (\!|\epsilon \in \alpha^1(\Sigma) \ ?\ \alpha^1(\Sigma') \ \text{\textcolon}\ \varnothing|\!) \ \text{\textcolon}\ \varnothing|\!) \ =\ \alpha^1(\Sigma) \oplus^1 \alpha^1(\Sigma') \\ & \qquad \wr \text{def. } \alpha^1 \text{ so that } \Sigma' \neq \varnothing \Longleftrightarrow \alpha^1(\Sigma') \neq \varnothing \text{ and def. } \oplus^1 \wr \\ - & \{a\} \oplus^1 F' \ =\ (\!|F' \neq \varnothing \ ?\ \{a\} \setminus \{\epsilon\} \cup (\!|\epsilon \in \{a\} \ ?\ F' \ \text{\textcolon}\ \varnothing|\!) \ \text{\textcolon}\ \varnothing|\!) \qquad \wr \text{def. } \oplus^1 \wr \\ = & (\!|F' \neq \varnothing \ ?\ \{a\} \ \text{\textcolon}\ \varnothing|\!) \qquad \wr \epsilon \notin \{a\} \wr. \quad \square \end{array}$$

The first concatenation is monotone (hence upper continuous since $\mathcal{T}$ is finite).

Lemma 73. *If $F_1 \subseteq F_1'$ and $F_2 \subseteq F_2'$ then $F_1 \oplus^1 F_2 \subseteq F_1' \oplus^1 F_2'$.* $\square$

Proof.

$$\begin{array}{lll}
& F_1 \oplus^1 F_2 = \llparenthesis F_2 \neq \varnothing \mathrel{?} (F_1 \setminus \{\epsilon\}) \cup \llparenthesis \epsilon \in F_1 \mathrel{?} F_2 \mathrel{\r{}} \varnothing \rrparenthesis \mathrel{\r{}} \varnothing \rrparenthesis & \wr\text{def. } \oplus^1 \text{ in Lemma 72}\wr \\
\subseteq & \llparenthesis F_2' \neq \varnothing \mathrel{?} (F_1 \setminus \{\epsilon\}) \cup \llparenthesis \epsilon \in F_1 \mathrel{?} F_2 \mathrel{\r{}} \varnothing \rrparenthesis \mathrel{\r{}} \varnothing \rrparenthesis & \wr\text{since } F_2 \subseteq F_2' \text{ so } F_2 \neq \varnothing \text{ implies } F_2' \neq \varnothing \text{ and } \cup \text{ is monotone}\wr \\
\subseteq & \llparenthesis F_2' \neq \varnothing \mathrel{?} (F_1 \setminus \{\epsilon\}) \cup \llparenthesis \epsilon \in F_1' \mathrel{?} F_2 \mathrel{\r{}} \varnothing \rrparenthesis \mathrel{\r{}} \varnothing \rrparenthesis & \wr\text{since } F_1 \subseteq F_1' \text{ so } \epsilon \in F_1 \text{ implies } \epsilon \in F_1' \text{ and } \cup \text{ is monotone}\wr \\
\subseteq & \llparenthesis F_2' \neq \varnothing \mathrel{?} (F_1' \setminus \{\epsilon\}) \cup \llparenthesis \epsilon \in F_1' \mathrel{?} F_2 \mathrel{\r{}} \varnothing \rrparenthesis \mathrel{\r{}} \varnothing \rrparenthesis & \wr\text{since } F_1 \subseteq F_1' \text{ so } (F_1 \setminus \{\epsilon\}) \subseteq (F_1' \setminus \{\epsilon\}) \text{ and } \cup \text{ is monotone}\wr \\
\subseteq & \llparenthesis F_2' \neq \varnothing \mathrel{?} (F_1' \setminus \{\epsilon\}) \cup \llparenthesis \epsilon \in F_1' \mathrel{?} F_2' \mathrel{\r{}} \varnothing \rrparenthesis \mathrel{\r{}} \varnothing \rrparenthesis & \wr\text{since } F_2 \subseteq F_2' \text{ and } \cup \text{ is monotone}\wr \\
= & F_1' \oplus^1 F_2' & \wr\text{def. } \oplus^1 \text{ in Lemma 72}\wr. \quad \square
\end{array}$$

The *first semantics* $\mathsf{S}^1[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\epsilon\})$ of a grammar $\mathcal{G}$ is

$$\mathsf{S}^1[\![\mathcal{G}]\!] \quad \triangleq \quad \dot{\alpha}^1(\mathsf{S}^\ell[\![\mathcal{G}]\!]). \tag{49}$$

The classical definition of the First derivation of a grammar [8, Def. 8.2.33] is

Theorem 74.

$$\mathsf{S}^1[\![\mathcal{G}]\!] \quad = \quad \boldsymbol{\lambda} A \bullet \{a \in \mathcal{T} \mid \exists \sigma \in \mathcal{T}^\star : A \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\epsilon \mid A \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\}. \quad \square$$

Proof. We calculate

$$\begin{array}{lll}
& \mathsf{S}^1[\![\mathcal{G}]\!] = \dot{\alpha}^1(\mathsf{S}^\ell[\![\mathcal{G}]\!]) = \boldsymbol{\lambda} A \bullet \alpha^1(\mathsf{S}^\ell[\![\mathcal{G}]\!](A)) & \wr\text{def. } \mathsf{S}^1[\![\mathcal{G}]\!] \text{ and } \dot{\alpha}^1\wr \\
= & \boldsymbol{\lambda} A \bullet \alpha^1(\{\sigma \in \mathcal{T}^\star \mid A \overset{\star}{\Longmapsto}_{\mathcal{G}} \sigma\}) & \wr\text{Theorem 71}\wr \\
= & \boldsymbol{\lambda} A \bullet \{a \in \mathcal{T} \mid \exists \sigma \in \mathcal{T}^\star : A \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\epsilon \mid A \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\} & \wr\text{def. } \alpha^1 \text{ and } \in\wr. \quad \square
\end{array}$$

The first semantics $\mathsf{S}^1[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\epsilon\})$ of a grammar $\mathcal{G}$ (49) can be extended to $\vec{\mathsf{S}}^1[\![\mathcal{G}]\!] \in \mathcal{V}^\star \mapsto \wp(\mathcal{T} \cup \{\epsilon\})$ as

$$\begin{array}{llllll}
\vec{\mathsf{S}}^1[\![\mathcal{G}]\!](\epsilon) & \triangleq & \{\epsilon\}, & \vec{\mathsf{S}}^1[\![\mathcal{G}]\!](a) & \triangleq & \{a\} \\
\vec{\mathsf{S}}^1[\![\mathcal{G}]\!](A) & \triangleq & \mathsf{S}^1[\![\mathcal{G}]\!](A) & \vec{\mathsf{S}}^1[\![\mathcal{G}]\!](\sigma_1\sigma_2) & \triangleq & \vec{\mathsf{S}}^1[\![\mathcal{G}]\!](\sigma_1) \oplus^1 \vec{\mathsf{S}}^1[\![\mathcal{G}]\!](\sigma_2)
\end{array} \tag{50}$$

so that

Theorem 75.

$$\vec{\mathsf{S}}^1[\![\mathcal{G}]\!] \quad = \quad \boldsymbol{\lambda} \sigma \bullet \{a \in \mathcal{T} \mid \exists \sigma' \in \mathcal{T}^\star : \sigma \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma'\} \cup \{\epsilon \mid \sigma = \epsilon\}. \quad \square$$

Proof. By induction on the length $|\sigma|$ of σ using Theorem 74 for nonterminals. $\square$

For parsing, the input sentence is often assumed to be followed by the final mark $\dashv$, so it is useful to extend $\mathsf{S}^1[\![\mathcal{G}]\!]$ to $\mathsf{S}^{1\dashv}[\![\mathcal{G}]\!] \in \mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\dashv\})$ as

$$\mathsf{S}^{1\dashv}[\![\mathcal{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda} A \bullet \{a \in \mathcal{T} \mid \exists \sigma \in \mathcal{T}^\star : A \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\dashv \mid A \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\}. \tag{51}$$

The first algorithm [32, Section 4.4] is indeed a fixpoint computation [8, Fig. 8.11] since $\mathsf{S}^1[\![\mathcal{G}]\!] = \mathbf{lfp}^{\dot{\subseteq}}\, \hat{\mathsf{F}}^1[\![\mathcal{G}]\!]$ where the bottom-up transformer $\hat{\mathsf{F}}^1[\![\mathcal{G}]\!]$ is (19) instantiated as given in Section 14.[7]

19.2. ϵ-productivity

The classical definition of ϵ-Prod [8, Section 8.2.3] provides information on which nonterminals can be empty. The corresponding abstraction is $\alpha^\epsilon \triangleq \boldsymbol{\lambda} \Sigma \bullet \llparenthesis \epsilon \in \Sigma \mathrel{?} \mathtt{tt} \mathrel{\r{}} \mathtt{ff} \rrparenthesis$ extended pointwise to $\dot{\alpha}^\epsilon \triangleq \boldsymbol{\lambda} L \bullet \boldsymbol{\lambda} A \bullet \alpha^\epsilon(L(A))$ so that

$$\langle \mathcal{N} \mapsto \wp(\mathcal{T}^\star),\ \dot{\subseteq} \rangle \quad \overset{\gamma^\epsilon}{\underset{\alpha^\epsilon}{\leftrightarrows}} \quad \langle \mathcal{N} \mapsto \mathbb{B},\ \dot{\Longrightarrow} \rangle.$$

The *ϵ-productivity semantics* $\mathsf{S}^\epsilon[\![\mathcal{G}]\!] \triangleq \dot{\alpha}^\epsilon(\mathsf{S}^\ell[\![\mathcal{G}]\!]) = \dot{\alpha}^\epsilon(\mathsf{S}^1[\![\mathcal{G}]\!])$ since $\dot{\alpha}^\epsilon = \dot{\alpha}^\epsilon \circ \dot{\alpha}^1$ and $\mathsf{S}^1[\![\mathcal{G}]\!] = \dot{\alpha}^1(\mathsf{S}^\ell[\![\mathcal{G}]\!])$. This is the classical definition of ϵ-productivity for a grammar [8, Section 8.2.9] since $\mathsf{S}^\epsilon[\![\mathcal{G}]\!] = \boldsymbol{\lambda} A \bullet A \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon$. The ϵ-productivity iterative computation [8, Fig. 8.14] is indeed a fixpoint computation $\mathsf{S}^\epsilon[\![\mathcal{G}]\!] = \mathbf{lfp}^{\dot{\Longrightarrow}}\, \hat{\mathsf{F}}^\epsilon[\![\mathcal{G}]\!]$ where the bottom-up transformer $\hat{\mathsf{F}}^\epsilon[\![\mathcal{G}]\!]$ is (19) instantiated as given in Section 14.

7 The classical definition [8, Fig. 8.11] is simpler since all grammar nonterminals are assumed to be productive.

19.3. Nonterminal productivity

The classical definition of *nonterminal productivity* [8, Section 8.2.4] provides information on which nonterminals of the grammar can produce a nonempty terminal language. The nonterminal productivity semantics of a context-free grammar is indeed an abstraction of its first semantics

$$\mathsf{S}^{\otimes}[\![\mathcal{G}]\!] \quad \triangleq \quad \dot{\alpha}^{\otimes}(\mathsf{S}^{\ell}[\![\mathcal{G}]\!]) \quad = \quad \dot{\alpha}^{\otimes}(\mathsf{S}^{1}[\![\mathcal{G}]\!]) \tag{52}$$

where the *nonterminal productivity abstraction* is defined pointwise on terminal languages derived for nonterminals $\dot{\alpha}^{\otimes} \triangleq \boldsymbol{\lambda} L \bullet \boldsymbol{\lambda} A \bullet \alpha^{\otimes}(L(A))$ as true if the nonterminal can produce a nonempty terminal language and false otherwise $\alpha^{\otimes} \triangleq \boldsymbol{\lambda}\, \Sigma \bullet (\!|\Sigma \neq \varnothing \,?\, \mathrm{tt} \,:\, \mathrm{ff}|\!)$ so that

$$\langle \mathcal{N} \mapsto \wp(\mathcal{T}^{\star}),\ \dot{\subseteq}\rangle \quad \xleftarrow[\dot{\alpha}^{\otimes}]{\dot{\gamma}^{\otimes}} \quad \langle \mathcal{N} \mapsto \mathbb{B},\ \dot{\Longrightarrow}\rangle.$$

The productivity iterative fixpoint computation [8, Ex. 8.2.12] is $\mathsf{S}^{\otimes}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\dot{\Longrightarrow}}\ \hat{\mathsf{F}}^{\otimes}[\![\mathcal{G}]\!]$ where the bottom-up transformer $\hat{\mathsf{F}}^{\otimes}[\![\mathcal{G}]\!]$ is (19) instantiated as given in Section 14.

The classical definition of productivity for a grammar nonterminal [8, Def. 8.2.5] is

Theorem 76.

$$\mathsf{S}^{\otimes}[\![\mathcal{G}]\!] \quad = \quad \boldsymbol{\lambda} A \bullet \exists \sigma \in \mathcal{T}^{\star} : A \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma. \quad \square$$

Proof. We calculate

$$\mathsf{S}^{\otimes}[\![\mathcal{G}]\!] = \dot{\alpha}^{\otimes}(\mathsf{S}^{\ell}[\![\mathcal{G}]\!]) = \dot{\alpha}^{\otimes}(\boldsymbol{\lambda} A \bullet \{\sigma \in \mathcal{T}^{\star} \mid A \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma\}) \qquad \wr \text{def. } \mathsf{S}^{\otimes}[\![\mathcal{G}]\!] \text{ and Theorem 71} \wr$$

$$= \quad \boldsymbol{\lambda} A \bullet \exists \sigma \in \mathcal{T}^{\star} : A \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma \qquad \wr \text{def. } \dot{\alpha}^{\otimes} \wr. \quad \square$$

Corollary 77. *We say that* all nonterminals of a grammar $\mathcal{G}$ are productive *if and only if* $\forall A \in \mathcal{N} : \mathsf{S}^{\otimes}[\![\mathcal{G}]\!](A) = \mathrm{tt}$*, in which case*

$$\forall \eta \in \mathcal{V}^{\star} : \exists \sigma \in \mathcal{T}^{\star} : \eta \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma. \quad \square$$

Proof. By induction over the length $|\eta|$ of η. By cases,

— if $\eta = a\eta'$ then $\exists \sigma \in \mathcal{T}^{\star} : \eta' \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma$ by induction hypothesis so $\eta = a\eta' \overset{\star}{\longmapsto}_{\mathcal{G}} a\sigma \in \mathcal{T}^{\star}$ by def. $\overset{\star}{\longmapsto}_{\mathcal{G}}$;

— if $\eta = A\eta'$ then $\exists \sigma \in \mathcal{T}^{\star} : A \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma$ by Theorem 76 and $\exists \sigma' \in \mathcal{T}^{\star} : \eta' \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma'$ by induction hypothesis so $\eta = A\eta' \overset{\star}{\longmapsto}_{\mathcal{G}} \sigma\sigma' \in \mathcal{T}^{\star}$ by def. $\overset{\star}{\longmapsto}_{\mathcal{G}}$;

— if $\eta = \epsilon$ then $\eta \overset{\star}{\longmapsto}_{\mathcal{G}} \epsilon \in \mathcal{T}^{\star}$ by def. $\overset{\star}{\longmapsto}_{\mathcal{G}}$. $\square$

20. Top-down grammar analysis

20.1. Follow grammar analysis

20.1.1. Follow

The classical definition of Follow [32, Section 4.4, p. 189], [8, Section 8.2.8] provides information on the possible right context of nonterminals during syntax analysis.

The *follow abstraction* $\alpha^{f} \in \mathcal{V}^{\star} \mapsto (\mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\dashv\}))$ is

$$\begin{aligned}\alpha^{f}(\eta) \quad \triangleq \quad & \boldsymbol{\lambda} A \bullet \{a \in \mathcal{T} \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \exists \eta''' \in \mathcal{T}^{\star} : \eta'' \overset{\star}{\longmapsto}_{\mathcal{G}} a\eta'''\} \cup \\ & \{\dashv \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \eta'' \overset{\star}{\longmapsto}_{\mathcal{G}} \epsilon\}\end{aligned}$$

where we use the classical convention that sentences derived from the grammar axiom $\overline{S}$ are assumed to be followed by the extra symbol $\dashv \notin \mathcal{V}$ ($\dashv$ is \$ in [32, Section 4.4] and # in [8, Section 8.2.8]). This is extended to $\alpha^{f}(\Sigma) \in \wp(\mathcal{V}^{\star}) \mapsto (\mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\dashv\}))$ as $\alpha^{f}(\Sigma) \triangleq \boldsymbol{\lambda} A \bullet \bigcup_{\eta \in \Sigma} \alpha^{f}(\eta)A$ so that

$$\langle \wp(\mathcal{V}^{\star}),\ \subseteq\rangle \quad \xleftarrow[\alpha^{f}]{\gamma^{f}} \quad \langle \mathcal{N} \mapsto \wp(\mathcal{T} \cup \{\dashv\}),\ \dot{\subseteq}\rangle.$$

The definition of Follow [32, Section 4.4, p. 189], [8, Def. 8.2.22] can also use that of First since

Theorem 78.

$$\alpha^f(\Sigma) \quad = \quad \lambda A \bullet \bigcup_{\eta' A \eta'' \in \Sigma} \vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\eta'')[\epsilon/\dashv] \quad \textit{where} \quad X[a/b] \;\triangleq\; (X \setminus \{a\}) \cup \{b \mid a \in X\}. \quad \square$$

Proof.

$$\alpha^f(\Sigma) \;=\; \lambda A \bullet \bigcup_{\eta\in\Sigma}\{a \in \mathscr{T} \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \exists \eta''' \in \mathscr{T}^\star : \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} a\eta'''\} \cup \{\dashv \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} \epsilon\}$$
$$\wr \text{def. } \alpha^f \wr$$

$$= \lambda A \bullet \bigcup_{\eta' A \eta'' \in \Sigma} (\vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\eta'') \setminus \{\epsilon\}) \cup \{\dashv \mid \epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\eta'')\} \qquad \wr\text{by Theorem 75}\wr$$

$$= \lambda A \bullet \bigcup_{\eta' A \eta'' \in \Sigma} \vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\eta'')[\epsilon/\dashv] \qquad \wr\text{by def. } X[a/b]\wr. \quad \square$$

20.1.2. Follow semantics

The *follow semantics* $\mathsf{S}^f[\![\mathscr{G}]\!]$ of a grammar $\mathscr{G}$ is

$$\mathsf{S}^f[\![\mathscr{G}]\!] \triangleq \alpha^f(\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!](\bar{S}))$$

so that we get [8, Def. 8.2.22]

Theorem 79.

$$\mathsf{S}^f[\![\mathscr{G}]\!] \quad = \quad \lambda A \bullet \{a \in \mathscr{T} \mid \exists \eta, \eta' : \bar{S} \overset{\star}{\Longmapsto}_{\mathscr{G}} \eta A a \eta'\} \cup \{\dashv \mid \exists \eta : \bar{S} \overset{\star}{\Longmapsto}_{\mathscr{G}} \eta A\}.$$

Proof.

$$\mathsf{S}^f[\![\mathscr{G}]\!] \;=\; \lambda A \bullet \bigcup\{\{a \in \mathscr{T} \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \exists \eta''' \in \mathscr{T}^\star : \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} a\eta'''\} \cup \{\dashv \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} \epsilon\} \mid \bar{S} \overset{\star}{\Longmapsto}_{\mathscr{G}} \eta\}$$
$$\wr\text{def. } \mathsf{S}^f[\![\mathscr{G}]\!], \text{ Corollary 59, and def. } \alpha^f\wr$$

$$= \lambda A \bullet \{a \in \mathscr{T} \mid \exists \eta, \eta' : \bar{S} \overset{\star}{\Longmapsto}_{\mathscr{G}} \eta A a \eta'\} \cup \{\dashv \mid \exists \eta : \bar{S} \overset{\star}{\Longmapsto}_{\mathscr{G}} \eta A\} \qquad \wr\text{def. } \cup \;\&\; \overset{\star}{\Longmapsto}_{\mathscr{G}}\wr. \quad \square$$

20.1.3. Fixpoint top-down structural follow semantics

By abstraction of the fixpoint characterization Theorem 57 of $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!]$, we get the classical Follow algorithm [32, Section 4.4, p. 189] as an iterative fixpoint computation [8, Fig. 8.13]

Theorem 80. $\mathsf{S}^f[\![\mathscr{G}]\!] \mathrel{\underline{\subseteq}} \mathbf{lfp}^{\dot{\subseteq}}\, \check{\mathsf{F}}^f[\![\mathscr{G}]\!]$ *where*

$$\check{\mathsf{F}}^f[\![\mathscr{G}]\!] \;\triangleq\; \lambda \phi \bullet \lambda A \bullet \{\dashv \mid A = \bar{S}\}$$
$$\cup \bigcup_{B \to \sigma A \sigma' \in \mathscr{R}} (\vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\sigma') \setminus \{\epsilon\}) \cup (\!|\epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\sigma') \;?\; \phi(B) \;\mathbin{⁏}\; \varnothing|\!)$$

and $\underline{\subseteq}$ *denotes* $=$ *if all nonterminals in* $\mathscr{G}$ *are productive (as defined in Section 19.3) else* $\underline{\subseteq}$ *denotes* $\subseteq$. $\square$

Proof. We have

$$= \quad \mathsf{S}^f[\![\mathscr{G}]\!] \quad = \quad \alpha^f \circ \alpha^{\bar{S}}(\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!])$$
$$\wr\text{def. } \mathsf{S}^f[\![\mathscr{G}]\!] \text{ where } \alpha^{\bar{S}} \text{ is the abstraction of functions at point } \bar{S} \text{ of Example 97}\wr$$

$$= \quad \alpha^f(\mathbf{lfp}^{\subseteq}\, \lambda X \bullet \{\bar{S}\} \cup \text{post}[\Longmapsto_{\mathscr{G}}]X) \qquad \wr\text{by Lemma 82}\wr$$

so that we apply Theorem 49 to this fixpoint definition.

— $$\alpha^f(\{\bar{S}\}) \quad = \quad \lambda A \bullet \{a \in \mathscr{T} \mid \exists \eta', \eta'' : \bar{S} = \eta' A \eta'' \wedge \exists \eta''' \in \mathscr{T}^\star : \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} a\eta'''\} \cup \{\dashv \mid \exists \eta', \eta'' : \bar{S} = \eta' A \eta'' \wedge \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} \epsilon\}$$
$$\wr\text{def. } \alpha^f\wr$$

$$= \quad \lambda A \bullet \{\dashv \mid A = \bar{S}\} \qquad \wr\text{def. sentence equality and } \overset{\star}{\Longmapsto}_{\mathscr{G}}\wr$$

— $\alpha^f(\text{post}[\Longmapsto_{\mathscr{G}}]X)$

$$\lambda A \bullet \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathscr{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \eta, \eta' : \eta \in X \wedge \eta \Longmapsto_{\mathscr{G}} \eta' A \eta''\} \qquad \wr\text{Theorem 78, def. } \cup, \text{ post}\wr$$

$= \ \boldsymbol{\lambda} A \bullet \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists n > 0, \varsigma_1, \ldots, \varsigma_{n+1}, A_1, \ldots, A_n, \sigma_1, \ldots, \sigma_n : \varsigma_1 A_1 \varsigma_2 \ldots \varsigma_n A_n \varsigma_{n+1} \in X \wedge \forall i \in [1, n] : A_i \to \sigma_i \in \mathscr{R} \wedge \exists \eta' : \eta' A \eta'' = \varsigma_1 \sigma_1 \varsigma_2 \ldots \varsigma_n \sigma_n \varsigma_{n+1}\}$ ≀def. (47) of $\Longrightarrow_{\mathcal{G}}$ and ∃≀

$= \ \boldsymbol{\lambda} A \bullet \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \varsigma_1, \varsigma_2, A_1, \sigma_1 : \varsigma_1 A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1 \in \mathscr{R} \wedge \exists \eta' : \eta' A \eta'' = \varsigma_1 \sigma_1 \varsigma_2\}$ ≀choosing $n = 1$≀

$= \ \boldsymbol{\lambda} A \bullet \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \varsigma_1', \varsigma_1'', \varsigma_2, A_1, \sigma_1 : \varsigma_1' A \varsigma_1'' A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1 \in \mathscr{R} \wedge \eta'' = \varsigma_1'' \sigma_1 \varsigma_2\} \cup \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \varsigma_1, \varsigma_2, A_1, \sigma_1', \sigma_1'' : \varsigma_1 A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R} \wedge \eta'' = \sigma_1'' \varsigma_2\} \cup \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \varsigma_1, \varsigma_2', A_1, \sigma_1 : \varsigma_1 A_1 \varsigma_2' A \eta'' \in X \wedge A_1 \to \sigma_1 \in \mathscr{R}\}$

≀since A must appear either in ς_1, σ_1 or ς_2≀

$= \ \boldsymbol{\lambda} A \bullet \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\left(\varsigma_1'' A_1 \varsigma_2\right)[\epsilon/\dashv] \mid \exists \varsigma_1', \sigma_1 : \varsigma_1' A \varsigma_1'' A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1 \in \mathscr{R}\} \cup \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\sigma_1'' \varsigma_2)[\epsilon/\dashv] \mid \exists \varsigma_1, A_1, \sigma_1' : \varsigma_1 A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R}\} \cup \bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \varsigma_1, \varsigma_2', A_1, \sigma_1 : \varsigma_1 A_1 \varsigma_2' A \eta'' \in X \wedge A_1 \to \sigma_1 \in \mathscr{R}\}$

≀by Theorem 74 so that $\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\varsigma_1'' A_1 \varsigma_2) = \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\varsigma_1'' \sigma_1 \varsigma_2)$ whenever $A_1 \to \sigma_1 \in \mathscr{R}$≀

— In this expression, let us first consider the term

$\bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\sigma_1'' \varsigma_2)[\epsilon/\dashv] \mid \exists \varsigma_1, A_1, \sigma_1' : \varsigma_1 A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R}\}$

$= \bigcup_{A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R}} \ \bigcup_{\varsigma_1 A_1 \varsigma_2 \in X} \left((\!|\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\varsigma_2 \neq \varnothing \ ?\ (\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\sigma_1'' \setminus \{\epsilon\}) \cup (\!|\epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!] \ ?\ \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\varsigma_2 \ \text{⨾}\ \varnothing|\!) \ \text{⨾}\ \varnothing|\!) \right) [\epsilon/\dashv]$

≀def. ∪, by (50), $\oplus^1$ in Lemma 72 and def. $\oplus^1$ in Lemma 72≀

$= \bigcup_{A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R}} \ \bigcup_{\varsigma_1 A_1 \varsigma_2 \in X} (\!|\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\varsigma_2 \neq \varnothing \ ?\ (\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\sigma_1'' \setminus \{\epsilon\}) \cup (\!|\epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!] \ ?\ \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\varsigma_2[\epsilon/\dashv] \ \text{⨾}\ \varnothing|\!) \ \text{⨾}\ \varnothing|\!)$

≀def. $X[a/b]$ so that $(X \setminus \{a\})[a/b] = (X \setminus \{a\})$, $\varnothing[a/b] = \varnothing$ and $X = \varnothing$ iff $X[a/b] = \varnothing$≀

$\subseteqq$ ≀$\subseteqq$ denotes $=$ if all nonterminals in $\mathcal{G}$ are productive (as defined in Section 19.3) in which case $\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\varsigma_2 \neq \varnothing$ else $\subseteqq$ denotes $\subseteq$≀

$\bigcup_{A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R}} \ \bigcup_{\varsigma_1 A_1 \varsigma_2 \in X} (\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\sigma_1'' \setminus \{\epsilon\}) \cup (\!|\epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\sigma_1'' \ ?\ \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\varsigma_2[\epsilon/\dashv] \ \text{⨾}\ \varnothing|\!)$

≀by Theorem 76 extended to protosentences≀

$= \bigcup_{A_1 \to \sigma_1' A \sigma_1'' \in \mathscr{R}} (\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\sigma_1'' \setminus \{\epsilon\}) \cup (\!|\epsilon \in \vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!]\sigma_1'' \ ?\ \alpha^f(X)A_1 \ \text{⨾}\ \varnothing|\!)$ ≀def. conditional and by Theorem 78;≀

— Second, in the above expression, the term

$\bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\varsigma_1'' A_1 \varsigma_2)[\epsilon/\dashv] \mid \exists \varsigma_1', \sigma_1 : \varsigma_1' A \varsigma_1'' A_1 \varsigma_2 \in X \wedge A_1 \to \sigma_1 \in \mathscr{R}\}$

is either $\varnothing$ or, by Theorem 78, is $\subseteq$-over approximated by $\alpha^f(X)A$;

— Third, and finally, in the above expression, the term

$\bigcup\{\vec{\mathsf{S}}^{\,1}[\![\mathcal{G}]\!](\eta'')[\epsilon/\dashv] \mid \exists \varsigma_1, \varsigma_2', A_1, \sigma_1 : \varsigma_1 A_1 \varsigma_2' A \eta'' \in X \wedge A_1 \to \sigma_1 \in \mathscr{R}\}$

is either $\varnothing$ or, by Theorem 78, is $\subseteq$-over approximated by $\alpha^f(X)A$.

It follows from the above calculation that

- If all nonterminals of $\mathcal{G}$ are productive, then

$$\check{\mathsf{F}}^f[\![\mathcal{G}]\!](\alpha^f(X)) \quad \subseteq \quad \alpha^f(\{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) \quad \subseteq \quad \check{\mathsf{F}}^f[\![\mathcal{G}]\!](\alpha^f(X)) \cup \alpha^f(X)$$

pointwise and so, by Corollary 101,

$$\mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^f[\![\mathcal{G}]\!] \quad \subseteq \quad \alpha^f(\mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) \quad \subseteq \quad \mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \check{\mathsf{F}}^f[\![\mathcal{G}]\!](X) \cup X$$

pointwise. By Example 103 applied pointwise, we have

$$\mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^f[\![\mathcal{G}]\!] \quad = \quad \mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \check{\mathsf{F}}^f[\![\mathcal{G}]\!](X) \cup X$$

so that $\alpha^f(\mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) = \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^f[\![\mathcal{G}]\!]$;

- Otherwise, we have

$$\alpha^f(\{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) \quad \subseteq \quad \check{\mathsf{F}}^f[\![\mathcal{G}]\!](\alpha^f(X)) \cup \alpha^f(X)$$

so that $\alpha^f(\mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) \subseteq \mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \check{\mathsf{F}}^f[\![\mathcal{G}]\!](X) \cup X = \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^f[\![\mathcal{G}]\!]$.

We conclude that $\alpha^f(\mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) \subseteqq \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^f[\![\mathcal{G}]\!]$, whence by equality or monotony, $\mathsf{S}^f[\![\mathcal{G}]\!] = \alpha^f(\mathbf{lfp}^{\subseteq}\, \boldsymbol{\lambda} X \bullet \{\bar{S}\} \cup \mathrm{post}[\Longmapsto_{\mathcal{G}}]X) \subseteqq \mathbf{lfp}^{\subseteq}\, \check{\mathsf{F}}^f[\![\mathcal{G}]\!]$. □

20.2. Nonterminal accessibility

The classical definition of *accessible nonterminals* [8, Def. 8.2.4] provides information on which nonterminals of the grammar are used in the definition of the language generated for the grammar axiom.

20.2.1. Accessibility abstraction of protosentences

The *accessibility abstraction* is defined on protolanguages as the characteristic function of the set of nonterminals appearing in the protosentences of this protolanguage

$$\alpha^a \quad \triangleq \quad \boldsymbol{\lambda}\, \Sigma \bullet \boldsymbol{\lambda}\, A \bullet (\!|\exists \sigma, \sigma' \in \mathscr{V}^\star : \sigma A \sigma' \in \Sigma \;?\; \mathtt{tt} \;\text{:}\; \mathtt{ff}|\!)$$

so that

$$\langle \mathscr{N} \mapsto \wp(\mathscr{V}^\star),\ \dot{\subseteq} \rangle \quad \underset{\alpha^a}{\overset{\gamma^a}{\leftrightarrows}} \quad \langle \mathscr{N} \mapsto \mathbb{B},\ \dot{\Longrightarrow} \rangle.$$

20.2.2. Accessibility semantics

The *nonterminal accessibility semantics* is

$$\mathsf{S}^a[\![\mathscr{G}]\!] \triangleq \alpha^a(\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!](\bar{S})) \quad = \quad \alpha^a \circ \alpha^{\bar{S}}(\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!])$$

where $\alpha^{\bar{S}}$ is the abstraction of functions at point $\bar{S}$ considered in Example 97.

This is the classical definition of productivity for a grammar nonterminal [8, Def. 8.2.4] since

Theorem 81.

$$\mathsf{S}^a[\![\mathscr{G}]\!] = \boldsymbol{\lambda}\, A \bullet \exists \sigma, \sigma' \in \mathscr{V}^\star : \bar{S} \overset{\star}{\longmapsto}_{\mathscr{G}} \sigma A \sigma'. \quad \square$$

Proof. We calculate $\mathsf{S}^a[\![\mathscr{G}]\!]$

$$\begin{aligned}
&= \boldsymbol{\lambda}\, A \bullet (\!|\exists \sigma, \sigma' \in \mathscr{V}^\star : \sigma A \sigma' \in \{\eta'' \in \mathscr{V}^\star \mid \bar{S} \overset{\star}{\longmapsto}_{\mathscr{G}} \eta''\} \;?\; \mathtt{tt} \;\text{:}\; \mathtt{ff}|\!) \\
&\qquad\qquad \wr \text{def. } \mathsf{S}^a[\![\mathscr{G}]\!], \text{Corollary 59, and def. } \alpha^a \wr \\
&= \boldsymbol{\lambda}\, A \bullet \exists \sigma, \sigma' \in \mathscr{V}^\star : \bar{S} \overset{\star}{\longmapsto}_{\mathscr{G}} \sigma A \sigma' \qquad \wr \text{def. } \in \wr. \quad \square
\end{aligned}$$

20.2.3. Fixpoint top-down structural accessibility semantics

We can project the top-down protolanguage semantics on a given nonterminal, in particular the start symbol $\bar{S}$, as follows

Lemma 82.

$$\alpha^{\bar{S}}(\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!]) \quad = \quad \boldsymbol{lfp}^{\subseteq}\ \boldsymbol{\lambda}\, X \bullet \{\bar{S}\} \cup \mathrm{post}[\longmapsto_{\mathscr{G}}]X. \quad \square$$

Proof. We have $\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!] = \mathbf{lfp}^{\subseteq}\ \check{\mathsf{F}}^{\check{L}}[\![\mathscr{G}]\!]$ where $\check{\mathsf{F}}^{\check{L}}[\![\mathscr{G}]\!] = \boldsymbol{\lambda}\, \phi \bullet \boldsymbol{\lambda}\, A \bullet f(A, \phi(A))$ with $f(A, X) = \{A\} \cup \mathrm{post}[\longmapsto_{\mathscr{G}}]X$ by Theorem 57 whence, by Example 105, $\alpha^{\bar{S}}(\mathsf{S}^{\check{L}}[\![\mathscr{G}]\!]) = \mathbf{lfp}^{\subseteq}\ \boldsymbol{\lambda}\, X \bullet \{\bar{S}\} \cup \mathrm{post}[\longmapsto_{\mathscr{G}}]X$. $\square$

The accessibility semantics $\mathsf{S}^a[\![\mathscr{G}]\!]$ has the following fixpoint characterization

Theorem 83.

$$\mathsf{S}^a[\![\mathscr{G}]\!] \quad = \quad \boldsymbol{lfp}^{\dot{\subseteq}}\ \check{\mathsf{F}}^a[\![\mathscr{G}]\!] \quad \textit{where} \quad \check{\mathsf{F}}^a[\![\mathscr{G}]\!] \quad \triangleq \quad \boldsymbol{\lambda}\, \phi \bullet \boldsymbol{\lambda}\, A \bullet (A = \bar{S}) \vee \bigvee_{B \to \sigma A \sigma' \in \mathscr{R}} \phi(B). \quad \square$$

Proof. Let us calculate $\alpha^a(\{\bar{S}\} \cup \mathrm{post}[\longmapsto_{\mathscr{G}}]X^\delta)$

$$\begin{aligned}
&= \boldsymbol{\lambda}\, A \bullet (A = \bar{S}) \vee \exists \eta \in X^\delta : \exists \eta', \eta'' : \eta \longmapsto_{\mathscr{G}} \eta' A \eta'' \\
&\qquad\qquad \wr \alpha^a \text{ preserves lubs, def. } \alpha^a \text{ and post} \wr \\
&= \boldsymbol{\lambda}\, A \bullet (A = \bar{S}) \vee \\
&\quad\ (\exists \eta \in X^\delta : \exists \eta', \eta'', \eta''', \eta'''', B \to \sigma \in \mathscr{R} : \eta = \eta' A \eta'' \wedge \eta = \eta''' B \eta'''') \vee \\
&\quad\ (\exists B \to \sigma A \sigma' \in \mathscr{R} : \alpha^a(X^\delta)(A)) \qquad \wr \text{def. } \alpha^a \wr
\end{aligned}$$

There are four possible cases for subformula

$$(\exists \eta \in X^\delta : \exists \eta', \eta'', \eta''', \eta'''', B \to \sigma \in \mathscr{R} : \eta = \eta' A \eta'' \wedge \eta = \eta''' B \eta''''), \tag{53}$$

as follows

1. $\forall \eta', \eta'' : \eta \neq \eta' A \eta''$, in which case (53) is false so $\alpha^a(\{\overline{S}\} \cup \text{post}[\Longmapsto_{\mathscr{G}}]X^\delta) = \check{\mathsf{F}}^a[\![\mathscr{G}]\!](\alpha^a(X^\delta))$ where $\check{\mathsf{F}}^a[\![\mathscr{G}]\!](\phi) \triangleq \boldsymbol{\lambda} A \bullet (A = \overline{S}) \vee (\exists B \to \sigma A \sigma' \in \mathscr{R} : \phi(B))$;
2. $\exists \eta', \eta'' : \eta = \eta' A \eta''$, with three subcases
   (a) neither η' nor η'' contains a nonterminal B so that $\forall \eta''', \eta'''' : \eta = \eta' A \eta'' \neq \eta''' B \eta''''$ in which case (53) is false so $\alpha^a(\{\overline{S}\} \cup \text{post}[\Longmapsto_{\mathscr{G}}]X^\delta) = \check{\mathsf{F}}^a[\![\mathscr{G}]\!](\alpha^a(X^\delta))$,
   (b) for all nonterminals B in either η' nor η'', their is no corresponding grammar rule $\forall \sigma : B \to \sigma \notin \mathscr{R}$ in which case (53) is true so $\alpha^a(\{\overline{S}\} \cup \text{post}[\Longmapsto_{\mathscr{G}}]X^\delta) = \check{\mathsf{F}}^a[\![\mathscr{G}]\!](\alpha^a(X^\delta))$,
   (c) either η' or η'' contains a nonterminal B such that $B \to \sigma \in \mathscr{R}$, in which case (53) is equal to $\overline{F}(\alpha^a(X^\delta))$ where $\overline{F}(\phi) \triangleq \boldsymbol{\lambda} A \bullet (A = \overline{S}) \vee \phi(A) \vee (\exists B \to \sigma A \sigma' \in \mathscr{R} : \phi(B))$.

Moreover $\check{\mathsf{F}}^a[\![\mathscr{G}]\!] \Longrightarrow \overline{F}$ pointwise, so

$$\check{\mathsf{F}}^a[\![\mathscr{G}]\!](\alpha^a(X^\delta)) \quad\Longrightarrow\quad \alpha^a(\{\overline{S}\} \cup \text{post}[\Longmapsto_{\mathscr{G}}]X^\delta) \quad\Longrightarrow\quad \overline{F}(\alpha^a(X^\delta))$$

and so by Corollary 101,

$$\mathbf{lfp}^{\Longrightarrow}\ \check{\mathsf{F}}^a[\![\mathscr{G}]\!] \quad\Longrightarrow\quad \alpha^a(\mathbf{lfp}^{\subseteq}\ \boldsymbol{\lambda} X \bullet \{\overline{S}\} \cup \text{post}[\Longmapsto_{\mathscr{G}}]X) \quad\Longrightarrow\quad \mathbf{lfp}^{\Longrightarrow}\ \overline{F}.$$

By Example 103 applied pointwise, we have $\mathbf{lfp}^{\Longrightarrow}\ \check{\mathsf{F}}^a[\![\mathscr{G}]\!] = \mathbf{lfp}^{\Longrightarrow}\ \overline{F}$ so by def. (53), Lemma 82 and antisymmetry, we conclude that $\mathsf{S}^a[\![\mathscr{G}]\!] = \mathbf{lfp}^{\Longrightarrow}\ \check{\mathsf{F}}^a[\![\mathscr{G}]\!]$. $\square$

The accessibility semantics of a context-free grammar is an abstraction of the follow semantics since, if all nonterminals are productive (as defined in Section 19.3), a nonterminal is accessible if and only if it has a nonempty follow set.

Theorem 84.

$$\textit{(All nonterminals are productive)} \quad\Longrightarrow\quad \big(\mathsf{S}^a[\![\mathscr{G}]\!] = \alpha^{\otimes}(\mathsf{S}^f[\![\mathscr{G}]\!])\big). \quad \square$$

Proof. Assuming all nonterminals to be productive, we prove that $\alpha^a = \dot{\alpha}^{\otimes} \circ \alpha^f$, as follows

$$\begin{aligned}
& \dot{\alpha}^{\otimes}(\alpha^f(\Sigma)) \\
= \ & \Big(\!\Big|\bigcup_{\eta \in \Sigma} \{a \in \mathscr{T} \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \exists \sigma \in \mathscr{T}^\star : \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} a\sigma\} \cup \{\dashv \mid \exists \eta', \eta'' : \eta = \eta' A \eta'' \wedge \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} \epsilon\} \neq \varnothing \mathrel{?} \mathtt{tt} \mathrel{\text{⸗}} \mathtt{ff}\Big|\!\Big) \\
& \hfill \wr \text{def. } \dot{\alpha}^{\otimes}, \alpha^{\otimes}, \text{and } \alpha^f \wr \\
= \ & (\!|\{\eta' A \eta'' \in \Sigma \mid \exists \sigma \in \mathscr{T}^\star : \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} \sigma\} \neq \varnothing \mathrel{?} \mathtt{tt} \mathrel{\text{⸗}} \mathtt{ff}|\!) \hfill \wr \text{def. } \mathscr{T}^\star \wr \\
= \ & (\!|\{\eta' A \eta'' \in \Sigma\} \neq \varnothing \mathrel{?} \mathtt{tt} \mathrel{\text{⸗}} \mathtt{ff}|\!) \hfill \wr \text{Corollary 77 so that } \exists \sigma \in \mathscr{T}^\star : \eta'' \overset{\star}{\Longmapsto}_{\mathscr{G}} \sigma \text{ by productivity hypothesis} \wr \\
= \ & \alpha^a(\Sigma) \hfill \wr \text{def. } \alpha^a \wr. \quad \square
\end{aligned}$$

21. Grammar problem

Knuth's *grammar problem* [1], a generalization of the single-source shortest-path problem, is to compute the minimum-cost derivation of a terminal string from each nonterminal of a given *superior grammar* that is a context-free grammar, with rules of the form $A \to g(A_1, \ldots, A_n)$, $n \geqslant 0$ (where 'g', '(', ',', and ')' are terminals), equipped with a cost function val such that the cost of a derivation is $\text{val}(A \to g(A_1, \ldots, A_n)) = \text{val}(g)(\text{val}(A_1), \ldots, \text{val}(A_n))$ and $\text{val}(g) \in \mathbb{R}_+^n \mapsto \mathbb{R}_+$, $\mathbb{R}_+ \triangleq \{x \in \mathbb{R} \mid x \geqslant 0\} \cup \{\infty\}$, is a so-called *superior* function [1], a condition weakened in [2] where Knuth's algorithm is also given an incremental version.

Knuth's grammar problem [1] can be generalized to any bottom-up abstract grammar semantics $\mathsf{S}^{\hat{\sharp}}[\![\mathscr{G}]\!]$ by considering $\alpha(\mathsf{S}^{\hat{\sharp}}[\![\mathscr{G}]\!])$ where $\langle \hat{\mathsf{D}}^{\hat{\sharp}}, \sqsubseteq \rangle \xleftrightarrow[\alpha]{\gamma} \langle \mathbb{R}_+, \geqslant \rangle$ is a Galois connection and $\langle \mathbb{R}_+, \geqslant, \infty, 0, \min, \max \rangle$ is a complete lattice.

Knuth considers the particular case when $\mathsf{S}^{\hat{\sharp}}[\![\mathscr{G}]\!] = \mathsf{S}^{\ell}[\![\mathscr{G}]\!]$ and $\langle \hat{\mathsf{D}}^{\hat{\sharp}}, \sqsubseteq \rangle = \langle \wp(S), \subseteq \rangle$ where S is a set (indeed $S = \wp(\mathscr{T}^\star)$ in [1,2]) with $\alpha(X) \triangleq \min\{\text{val}(x) \mid x \in X\}$ and $\gamma(m) \triangleq \{x \in S \mid \text{val}(x) \geqslant m\}$. Since α is antitone, the corresponding abstract semantics is taken in terms of greatest fixpoints for $\leqslant$ [2]. Knuth's monotony hypothesis [1,2] ensures the existence of the greatest fixpoint. The rule soundness and completeness condition (23) then amounts to Knuth's hypothesis that for every nonterminal A, every string in $\mathsf{S}^{\ell}[\![\mathscr{G}]\!]A$ is a composition of superior functions $\alpha(g(x_1, \ldots, x_n)) = \text{val}(g)(\alpha(x_1), \ldots, \alpha(x_n))$.

Knuth superiority condition [1] and its variant [2] ensure that the greatest fixpoint can be computed by an elimination algorithm (generalizing Dijkstra's algorithm to solve shortest-path problems [33]). However in general one must resort to an infinite fixpoint iteration as shown with the choice of $S = \wp(\mathscr{T}^\star)$, $\text{val}(x) = \frac{1}{|x|}$ so that $\text{val}(g)() = \frac{1}{3}$ and $\text{val}(g)(x_1, \ldots, x_n) = \frac{1}{\frac{1}{x_1} + \ldots + \frac{1}{x_n} + n + 2}$ which, for the grammar $A \to a()$, $A \to b(A, A)$ requires an infinite iteration and a passage to the limit 0.

Our generalization also copes with implicit abstractions of a grammar considered by [1,2] where a grammar is "recoded" into a superior grammar, which can indeed be defined by an appropriate α.

22. Bottom-up parsing

Given a grammar $\mathcal{G} = \langle \mathcal{T}, \mathcal{N}, \overline{S}, \mathcal{R}\rangle$ and an input $\sigma = \sigma_1\sigma_2\ldots\sigma_n \in \mathcal{T}^\star$, $n \geq 0$, *parsing* consists in proving either $\sigma \in \mathsf{S}^\ell[\![\mathcal{G}]\!](\overline{S})$ or $\sigma \notin \mathsf{S}^\ell[\![\mathcal{G}]\!](\overline{S})$, that is, by Theorem 71, providing an algorithmic answer to the question $\overline{S} \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \sigma$?

Bottom-up parsing is an abstraction of a bottom-up grammar semantics by restriction to a given input sentence. This is illustrated with the Cocke–Younger–Kasami or CYK algorithm [4, Section 4.2.1] attributed by [34] to John Cocke, [35,36]). It is traditionally restricted to grammars $\mathcal{G} = \langle \mathcal{T}, \mathcal{N}, \overline{S}, \mathcal{R}\rangle$ in *Chomsky normal form* with rules of the form $A \to BC$ and $A \to a$ where $A, B, C \in \mathcal{N}$ and $a \in \mathcal{T}$. We now design CYK by calculus for arbitrary grammars.

22.1. The concrete semantics and its abstraction

CYK is an abstract interpretation of the terminal language semantics $\mathsf{S}^\ell[\![\mathcal{G}]\!]$ (34) by

$$\alpha^{CYK} \quad \stackrel{\Delta}{=} \quad \boldsymbol{\lambda}\,\sigma \bullet \boldsymbol{\lambda}\,X \bullet \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i \ldots \sigma_{i+j-1} \in X\} \tag{54}$$

where

$$\hat{\mathsf{D}}^{CYK} \quad \stackrel{\Delta}{=} \quad \boldsymbol{\lambda}\,\sigma \bullet \{\langle i,\ j\rangle \mid i \in [1, |\sigma|+1] \wedge j \in [0, |\sigma|] \wedge i+j \leq |\sigma|+1\}$$

so that $\langle i,\ j\rangle$ denotes the subsentence of length j from position i in σ (in particular $\langle |\sigma|+1,\ 0\rangle$ denotes the empty sentence ϵ after $\sigma = \sigma\epsilon$). Given $\sigma \in \mathcal{T}^\star$, we have

$$\langle \wp(\mathcal{T}^\star),\ \subseteq\rangle \quad \underset{\alpha^{CYK}(\sigma)}{\overset{\gamma^{CYK}(\sigma)}{\leftrightarrows}} \quad \langle \wp(\hat{\mathsf{D}}^{CYK}(\sigma)),\ \subseteq\rangle.$$

The pointwise extension to $\mathcal{N}$ is

$$\alpha^{CYK} \quad \stackrel{\Delta}{=} \quad \boldsymbol{\lambda}\,\sigma \bullet \boldsymbol{\lambda}\,X \bullet \boldsymbol{\lambda}\,A \bullet \alpha^{CYK}(X(A)) \tag{55}$$

so that

$$\langle \mathcal{N} \mapsto \wp(\mathcal{T}^\star),\ \dot{\subseteq}\rangle \quad \underset{\alpha^{CYK}(\sigma)}{\overset{\gamma^{CYK}(\sigma)}{\leftrightarrows}} \quad \langle \mathcal{N} \mapsto \wp(\hat{\mathsf{D}}^{CYK}(\sigma)),\ \dot{\subseteq}\rangle.$$

22.2. Soundness of the parser

The correctness of this parsing approach is proved by the following

Theorem 85. $\sigma \in \mathsf{S}^\ell[\![\mathcal{G}]\!](\overline{S}) \Longleftrightarrow \langle 1,\ |\sigma|\rangle \in \alpha^{CYK}(\sigma)(\mathsf{S}^\ell[\![\mathcal{G}]\!])(\overline{S})$. $\square$

Proof. $\langle 1,\ |\sigma|\rangle \in \alpha^{CYK}(\sigma)(\mathsf{S}^\ell[\![\mathcal{G}]\!])(\overline{S})$

$\Longleftrightarrow \quad \langle 1,\ |\sigma|\rangle \in \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i \ldots \sigma_{i+j-1} \in \mathsf{S}^\ell[\![\mathcal{G}]\!](\overline{S})\}$ $\quad$ ≀def. (55) of α^{CYK}≀

$\Longleftrightarrow \quad \sigma \in \mathsf{S}^\ell[\![\mathcal{G}]\!](\overline{S})$ $\quad$ ≀def. $\in$ and $\sigma_1 \ldots \sigma_{|\sigma|}$≀. $\square$

22.3. Design of the parser

The CYK algorithm is derived by abstracting the fixpoint definition Theorem 44 of $\mathsf{S}^\ell[\![\mathcal{G}]\!] = \mathbf{lfp}^{\dot{\subseteq}}\,\hat{\mathsf{F}}^\ell[\![\mathcal{G}]\!]$ by α^{CYK}.

Theorem 86.

$$\alpha^{CYK}(\sigma)(\mathsf{S}^\ell[\![\mathcal{G}]\!])(\overline{S}) \quad = \quad \mathbf{lfp}^{\dot{\subseteq}}\,\hat{\mathsf{F}}^{CYK}[\![\mathcal{G}]\!](\sigma)$$

where

$$\begin{array}{rcl}
\hat{\mathsf{F}}^{CYK}[\![\mathcal{G}]\!] & \in & \wp(\hat{\mathsf{D}}^{CYK}) \mapsto \wp(\hat{\mathsf{D}}^{CYK}) \\
\hat{\mathsf{F}}^{CYK}[\![\mathcal{G}]\!] & \stackrel{\Delta}{=} & \boldsymbol{\lambda}\,\rho \bullet \boldsymbol{\lambda}\,A \bullet \displaystyle\bigcup_{A\to\sigma\in\mathcal{R}} \hat{\mathsf{F}}^{CYK}[A \to \boldsymbol{.}\sigma]\rho \\
\hat{\mathsf{F}}^{CYK}[A \to \sigma\boldsymbol{.}a\sigma'] & \stackrel{\Delta}{=} & \boldsymbol{\lambda}\,\rho \bullet \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i = a \wedge \\
& & \qquad \langle i+1,\ j-1\rangle \in \hat{\mathsf{F}}^{CYK}[A \to \sigma a\boldsymbol{.}\sigma']\rho\} \\
\hat{\mathsf{F}}^{CYK}[A \to \sigma\boldsymbol{.}B\sigma'] & \stackrel{\Delta}{=} & \boldsymbol{\lambda}\,\rho \bullet \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \exists k : 0 \leqslant k \leqslant j : \langle i,\ k\rangle \in \rho(B) \\
& & \qquad \wedge \langle i+k,\ j-k\rangle \in \hat{\mathsf{F}}^{CYK}[A \to \sigma B\boldsymbol{.}\sigma']\rho\} \\
\hat{\mathsf{F}}^{CYK}[A \to \sigma\boldsymbol{.}] & \stackrel{\Delta}{=} & \boldsymbol{\lambda}\,\rho \bullet \{\langle i,\ 0\rangle \mid 1 \leqslant i \leqslant |\sigma|\} \quad \square
\end{array}$$

Proof. We apply Corollary 106.

$$\begin{aligned}
& \alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!](\rho)) \\
= \ & \alpha^{CYK}(\sigma)\left(\lambda A \bullet \bigcup_{A\to\sigma\in\mathcal{R}} \hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \bullet\sigma]\rho\right) && \wr\text{def. (35) of } \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!]\wr \\
= \ & \left\{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i \ldots \sigma_{i+j-1} \in \bigcup_{A\to\sigma\in\mathcal{R}} \hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \bullet\sigma]\rho\right\} && \wr\text{def. (55) of } \alpha^{CYK}\wr \\
= \ & \bigcup_{A\to\sigma\in\mathcal{R}} \alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \bullet\sigma]\rho) && \wr\text{def. } \in \text{ and (54) of } \alpha^{CYK}\wr \\
= \ & \bigcup_{A\to\sigma\in\mathcal{R}} \hat{\mathsf{F}}^{CYK}[A \to \bullet\sigma](\alpha^{CYK}(\sigma)(\rho))
\end{aligned}$$

$\wr$provided we can define $\hat{\mathsf{F}}^{CYK}$ such that $\alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\bullet\sigma']\rho) = \hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet\sigma'](\alpha^{CYK}(\sigma)(\rho))\wr$.

We proceed by induction on the length $|\sigma'|$ of σ', with three cases.

$$\begin{aligned}
& \alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\bullet a\sigma']\rho) \quad = \quad \alpha^{CYK}(\sigma)(a\,\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma a\bullet\sigma']\rho) && \wr\text{def. } \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!]\wr \\
= \ & \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i \ldots \sigma_{i+j-1} \in (a\,\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma a\bullet\sigma']\rho)\} && \wr\text{def. (54) of } \alpha^{CYK}\wr \\
= \ & \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i = a \wedge \langle i+1,\ j-1\rangle \in \alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma a\bullet\sigma']\rho)\} && \wr\text{def. concat., } \in\text{, and (54) of } \alpha^{CYK}\wr \\
= \ & \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i = a \wedge \langle i+1,\ j-1\rangle \in \hat{\mathsf{F}}^{CYK}[A \to \sigma a\bullet\sigma'](\alpha^{CYK}(\sigma)(\rho))\} && \wr\text{ind. hyp.}\wr \\
= \ & \hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet a\sigma'](\alpha^{CYK}(\sigma)(\rho))
\end{aligned}$$

$\wr$by defining $\hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet a\sigma']\rho \triangleq \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i = a \wedge \langle i+1,\ j-1\rangle \in \hat{\mathsf{F}}^{CYK}[A \to \sigma a\bullet\sigma']\rho\}\wr$

$$\begin{aligned}
& \alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\bullet B\sigma']\rho) \quad = \quad \alpha^{CYK}(\sigma)(\rho(B)\,\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma B\bullet\sigma']\rho) && \wr\text{def. } \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!]\wr \\
= \ & \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i \ldots \sigma_{i+j-1} \in (\rho(B)\,\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma B\bullet\sigma']\rho)\} && \wr\text{def. (54) of } \alpha^{CYK}\wr \\
= \ & \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \exists k : 0 \leqslant k \leqslant j : \langle i,\ k\rangle \in \alpha^{CYK}(\rho)(B) \wedge \langle i+k,\ j-k\rangle \in \alpha^{CYK}(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma B\bullet\sigma']\rho)\} \\
& && \wr\text{def. concatenation, (54) and (55) of } \alpha^{CYK}\wr \\
= \ & \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \exists k : 0 \leqslant k \leqslant j : \langle i,\ k\rangle \in \alpha^{CYK}(\rho)(B) \wedge \langle i+k,\ j-k\rangle \in \hat{\mathsf{F}}^{CYK}[A \to \sigma B\bullet\sigma'](\alpha^{CYK}(\rho))\} && \wr\text{ind. hyp.}\wr \\
= \ & \hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet B\sigma'](\alpha^{CYK}(\sigma)(\rho))
\end{aligned}$$

$\wr$by defining $\hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet B\sigma']\rho \triangleq \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \exists k : 0 \leqslant k \leqslant j : \langle i,\ k\rangle \in \rho(B) \wedge \langle i+k,\ j-k\rangle \in \hat{\mathsf{F}}^{CYK}[A \to \sigma B\bullet\sigma']\rho\}\wr$

$$\begin{aligned}
& \alpha^{CYK}(\sigma)(\hat{\mathsf{F}}^{\ell}_{\cdot}[A \to \sigma\bullet]\rho) \quad = \quad \{\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma) \mid \sigma_i \ldots \sigma_{i+j-1} = \epsilon\} \\
& && \wr\text{def. } \hat{\mathsf{F}}^{\ell}[\![\mathcal{G}]\!] \text{ and (54) of } \alpha^{CYK}\wr \\
= \ & \{\langle i,\ 0\rangle \mid 1 \leqslant i \leqslant |\sigma|\} \quad = \quad \hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet](\alpha^{CYK}(\sigma)(\rho))
\end{aligned}$$

$\wr$def. equality of sentences and by defining $\hat{\mathsf{F}}^{CYK}[A \to \sigma\bullet]\rho \triangleq \{\langle i,\ 0\rangle \mid 1 \leqslant i \leqslant |\sigma|\}\wr$. □

The original CYK algorithm is only defined for grammars in CNF (Chomsky Normal Form) whence we get a generalization to arbitrary context-free grammars.

22.4. Parsing algorithm

Because the abstract domain $\langle \mathcal{N} \mapsto \wp(\hat{\mathsf{D}}^{CYK}(\sigma)),\ \dot{\subseteq}\rangle$ is finite, the iterative computation of $\mathbf{lfp}^{\dot{\subseteq}}\ \mathsf{F}^{CYK}[\![\mathcal{G}]\!](\sigma)$ terminates whence by Theorems 85 and 86 so does the CYK parsing algorithm. The CYK dynamic programming algorithm organizes the computation of the pairs $\langle i,\ j\rangle \in \hat{\mathsf{D}}^{CYK}(\sigma)$ in order to avoid repetition of work already done.

23. Top-down parsing

23.1. Nonrecursive predictive parser

The general idea of the formal derivation of parsers by abstract interpretation is that a parser is an abstraction of a grammar semantics by restriction of this semantics to a given input sentence.

A nonrecursive predictive parser is formally derived from the prefix derivation semantics $\vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!]$ of Section 6 by applying this idea with the abstraction

$$\alpha^{LL} \triangleq \boldsymbol{\lambda}\, \overline{S} \bullet \boldsymbol{\lambda}\, \sigma \bullet \boldsymbol{\lambda}\, X \bullet \{\langle i,\ \varpi\rangle \mid \exists \theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\overline{S} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m\}$$

where the *terminal abstraction* $\alpha^{\tau} \in \Theta \mapsto \mathscr{T}^{\star}$ collects terminal labels of derivations, as follows

$$\begin{array}{llll} \alpha^{\tau}(\theta_1 \xrightarrow{(\!|A} \theta_2) \triangleq \alpha^{\tau}(\theta_1)\alpha^{\tau}(\theta_2) & & \alpha^{\tau}(\varpi) \triangleq \epsilon, & \varpi \in \mathscr{S} \\ \alpha^{\tau}(\theta_1 \xrightarrow{A|\!)} \theta_2) \triangleq \alpha^{\tau}(\theta_1)\alpha^{\tau}(\theta_2) & & \alpha^{\tau}(\vdash) \triangleq \epsilon & \\ \alpha^{\tau}(\theta_1 \xrightarrow{a} \theta_2) \triangleq \alpha^{\tau}(\theta_1)a\alpha^{\tau}(\theta_2), & a \in \mathscr{T} & \alpha^{\tau}(\dashv) \triangleq \epsilon. & \end{array}$$

The interpretation of the pair $\langle i,\ \varpi_i\rangle$ is that in the left-to-right scanning of the input sentence σ up to position i, the prefix $\sigma_1 \ldots \sigma_i$ (ϵ when $i = 0$) has been recognized by a prefix derivation from the start symbol $\overline{S}$. The stack ϖ allows for the recognition of the rest of the sentence, if possible.

Let us write $\wp_1(S) \triangleq \{\{x\} \mid x \in S\}$ for the set of singletons of a set S and let $\alpha^{\bullet} \in \wp_1(S) \mapsto S$ be $\alpha^{\bullet}(\{x\}) \triangleq x$. We have

Lemma 87.

$$\forall \theta \in \Theta_{()} : \alpha^{\tau}(\theta) \quad = \quad \alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\theta). \quad \square$$

Proof. Given a well-parenthesized trace $\theta \in \Theta_{()}$, we prove, by induction on the length of traces, that $\alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\theta)$ is a singleton and $\alpha^{\tau}(\theta) = \alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\theta)$. We proceed by cases.

— If θ contains parentheses in $\mathscr{P}$, then by definition of well-parenthesized traces in Section 5.4, the trace θ must have the form $\theta = \theta_1\varpi_1 \xrightarrow{(\!|A} \theta_2 \xrightarrow{A|\!)} \varpi_2\theta_3$ where $(\!|A$ and $A|\!)$ are matching parentheses. Therefore

$\alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\theta_1\varpi_1 \xrightarrow{(\!|A} \theta_2 \xrightarrow{A|\!)} \varpi_2\theta_3)$

$= \alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1))\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\varpi_1))(\!|A\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_2))A|\!)\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\varpi_2))\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_3)))$
$\wr$def. $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathscr{D}}$ and $\alpha^{\hat{s}} \in \hat{\mathscr{D}} \mapsto \hat{\mathscr{T}}$$\wr$

$= \alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1))(\!|A\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_2))A|\!)\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_3)))$
$\wr$def. $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathscr{D}}$ and $\alpha^{\hat{s}} \in \hat{\mathscr{D}} \mapsto \hat{\mathscr{T}}$ so that $\forall \varpi \in \mathscr{S} : \alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\varpi)) = \epsilon$$\wr$

$= \alpha^{\bullet} \circ \alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1)))(\{A\} \cup \alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_2))))\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_3))))$ $\wr$def. $\alpha^{\hat{L}}$ and well-parenthesization hypothesis$\wr$

$= \alpha^{\bullet}(\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1))))\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_2))))\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_3)))))$ $\wr$def. α^{ℓ} and, by ind. hyp., concatenation of singletons which is a singleton$\wr$

$= \alpha^{\bullet}(\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1)))))\alpha^{\bullet}(\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_2)))))\alpha^{\bullet}(\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_3)))))$ $\wr$def. $\alpha^{\bullet}$ and concatenation of singletons$\wr$

$= \alpha^{\tau}(\theta_1)\alpha^{\tau}(\theta_2)\alpha^{\tau}(\theta_3)$ $\wr$by ind. hyp.$\wr$

$= \alpha^{\tau}(\theta_1\varpi_1 \xrightarrow{(\!|A} \theta_2 \xrightarrow{A|\!)} \varpi_2\theta_3)$ $\wr$def. $\alpha^{\tau}$$\wr$

— $\alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\theta_1 \xrightarrow{a} \theta_2)$

$= \alpha^{\bullet} \circ \alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1)))\{a\}\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_2))))$
$\wr$def. $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathscr{D}}$, $\alpha^{\hat{s}} \in \hat{\mathscr{D}} \mapsto \hat{\mathscr{T}}$, and $\alpha^{\hat{L}}$ since θ_2 and θ_2 are well-parenthesized$\wr$

$= \alpha^{\bullet}(\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta_1)))))a\alpha^{\bullet}(\alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\theta)))))$ $\wr$def. α^{ℓ} and $\alpha^{\bullet}$ and concatenation of singletons by ind. hyp.$\wr$

$= \alpha^{\tau}(\theta_1)a\alpha^{\tau}(\theta_2) \quad = \quad \alpha^{\tau}(\theta_1 \xrightarrow{a} \theta_2)$ $\wr$by ind. hyp. and def. $\alpha^{\tau}$$\wr$

— $\alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\vdash) \quad = \quad \alpha^{\bullet}\left(\bigcup\{\{\epsilon\}\}\right)$ $\wr$def. $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathscr{D}}$, $\alpha^{\hat{s}} \in \hat{\mathscr{D}} \mapsto \hat{\mathscr{T}}$, $\alpha^{\hat{L}} \in \hat{\mathscr{T}} \mapsto \wp(\mathscr{V}^{\star})$, $\alpha^{\ell} \in \wp(\mathscr{V}^{\star}) \mapsto \wp(\mathscr{T}^{\star})$, and $\alpha^{\ell} \in \mathscr{V}^{\star} \mapsto \wp(\mathscr{T}^{\star})$$\wr$

$= \alpha^{\bullet}(\{\epsilon\}) \quad = \quad \epsilon \quad = \quad \alpha^{\tau}(\vdash)$ $\wr$def. $\bigcup$, $\alpha^{\bullet}$, and $\alpha^{\tau}$$\wr$

— $\alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}(\dashv) \quad = \quad \alpha^{\bullet} \circ \alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}}(\epsilon)$ $\wr$def. $\alpha^{\hat{\delta}} \in \Theta \mapsto \hat{\mathscr{D}}$$\wr$

$= \alpha^{\tau}(\dashv) \quad = \quad \epsilon$ $\wr$as shown above and by def. $\alpha^{\tau}$$\wr$. $\square$

Fixing the start symbol $\overline{S}$ and the input sentence σ, we have a Galois connection

$$\langle \wp(\Theta),\ \subseteq\rangle \quad \underset{\alpha^{LL}(\overline{S})(\sigma)}{\overset{\gamma^{LL}(\overline{S})(\sigma)}{\leftrightarrows}} \quad \langle \wp([0, |\sigma|] \times \mathscr{S}),\ \subseteq\rangle.$$

The correctness of this parsing approach is proved by the following

Theorem 88.

$$\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S}) \quad\Longleftrightarrow\quad \langle|\sigma|,\ \dashv\rangle \in \alpha^{LL}(\bar{S})(\sigma)(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!]). \quad \square$$

Proof. We calculate $\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S})$

$\Longleftrightarrow\quad \sigma \in \alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].\bar{S}))))$
 $\wr$def. (34) of $\mathsf{S}^{\ell}[\![\mathcal{G}]\!]$, (31) of $\mathsf{S}^{\hat{L}}[\![\mathcal{G}]\!]$, (29) of $\mathsf{S}^{\hat{s}}[\![\mathcal{G}]\!]$, (26) of $\mathsf{S}^{\hat{\delta}}[\![\mathcal{G}]\!]$, $\dot{\alpha}^{\ell}$, Section 13.3.3 of $\dot{\alpha}^{\hat{L}}$, $\alpha^{\hat{s}}$, $\alpha^{\hat{\delta}}$ and selection $\bullet.\bar{S}$ $\wr$

$\Longleftrightarrow\quad \exists\theta \in \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].\bar{S} : \sigma \in \alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\{\theta\}))))$
 $\wr$since $\alpha^{\ell} \circ \alpha^{\hat{L}} \circ \alpha^{\hat{s}} \circ \alpha^{\hat{\delta}}$ is the lower adjoint of a composition of Galois connections whence of a Galois connection, whence preserves lubs hence $\sigma \in \alpha(X) = \alpha(\bigcup_{x\in X}\{x\}) = \bigcup_{x\in X}\alpha(\{x\})$ if and only if $\exists x \in X : \sigma \in \alpha(\{x\})$$\wr$

$\Longleftrightarrow\quad \exists\theta \in \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].\bar{S} : \sigma = \alpha^{\bullet} \circ \alpha^{\ell}(\alpha^{\hat{L}}(\alpha^{\hat{s}}(\alpha^{\hat{\delta}}(\{\theta\}))))$
 $\wr$def. $\alpha^{\bullet}$ and the image of a singleton by α^{ℓ}, $\alpha^{\hat{L}}$, $\alpha^{\hat{s}}$ or $\alpha^{\hat{\delta}}$ is a singleton$\wr$

$\Longleftrightarrow\quad \exists\theta \in \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].\bar{S} : \alpha^{\tau}(\theta) = \sigma$ $\wr$Lemma 87$\wr$

$\Longleftrightarrow\quad \exists\theta \in (\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!].\bar{S} \cap \Theta^{\dashv}) : \alpha^{\tau}(\theta) = \sigma$ $\wr$by (6) so that $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!] = \mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!] \cap \Theta^{\dashv}$ & def. $\bullet.\bar{S}$$\wr$

$\Longleftrightarrow\quad \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!].\bar{S} \cap \Theta^{\dashv} : \alpha^{\tau}(\theta) = \sigma$ $\wr$def. (5) of $\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!]$ (so that $\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m$)$\wr$

$\Longleftrightarrow\quad \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \dashv \in \mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!].\bar{S} : \alpha^{\tau}(\theta) = \sigma$ $\wr$since $\varpi_m = \dashv$ by def. $\Theta^{\dashv}$$\wr$

$\Longleftrightarrow\quad \langle|\sigma|,\ \dashv\rangle \in \{\langle i,\ \varpi'\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!].\bar{S} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi' = \varpi_m\}$
 $\wr$since $\sigma = \sigma_1 \ldots \sigma_{|\sigma|}$ and def. $\in$$\wr$

$\Longleftrightarrow\quad \langle|\sigma|,\ \dashv\rangle \in \alpha^{LL}(\bar{S})(\sigma)(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!])$ $\wr$def. $\alpha^{LL}(\bar{S})(\sigma)$$\wr$. $\square$

To get a correct parsing algorithm, it remains

- to express $\alpha^{LL}(\bar{S})(\sigma)(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!])$ in fixpoint form by abstraction of the fixpoint definition Theorem 8 of $\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!]$ (as shown in Theorem 89), and
- to prove the termination of the fixpoint iteration (as shown in Theorem 91 for non-left-recursive grammars).

Theorem 89.

$$\alpha^{LL}(\bar{S})(\sigma)(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!]) \quad = \quad \boldsymbol{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$$

where

$$\begin{aligned}
&\mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma) \in \wp([0, |\sigma|] \times \mathcal{S}) \mapsto \wp([0, |\sigma|] \times \mathcal{S})\\
&\mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma) = \boldsymbol{\lambda} X \bullet \{\langle 0,\ \vdash\rangle\} \cup \{\langle 0,\ \dashv[\bar{S} \to \bullet\eta]\rangle \mid \langle 0,\ \vdash\rangle \in X \wedge \bar{S} \to \eta \in \mathcal{R}\}\\
&\qquad \cup \{\langle i+1,\ \varpi[A \to \eta a\bullet\eta']\rangle \mid \langle i,\ \varpi[A \to \eta\bullet a\eta']\rangle \in X \wedge a = \sigma_{i+1}\}\\
&\qquad \cup \{\langle i,\ \varpi[A \to \eta B\bullet\eta'][B \to \bullet\varsigma]\rangle \mid \langle i,\ \varpi[A \to \eta\bullet B\eta']\rangle \in X \wedge B \to \varsigma \in \mathcal{R}\}\\
&\qquad \cup \{\langle i,\ \varpi\rangle \mid \langle i,\ \varpi[A \to \eta\bullet]\rangle \in X\}. \quad \square
\end{aligned}$$

Proof. We use the fixpoint characterization of $\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!]$ in Theorem 8 as $\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!] = \mathbf{lfp}^{\subseteq}\ \mathsf{F}^{\vec{\partial}}[\![\mathcal{G}]\!]$ and apply the commutation condition to the transformer $\mathsf{F}^{\vec{\partial}}[\![\mathcal{G}]\!] \triangleq \boldsymbol{\lambda} X \bullet \{\vdash\} \cup X \fatsemi \longrightarrow$. Assuming X to be an iterate of $\mathsf{F}^{\vec{\partial}}[\![\mathcal{G}]\!]$, we calculate $\alpha^{LL}(\bar{S})(\sigma)(\{\vdash\} \cup X \fatsemi \longrightarrow)$

$= \quad \alpha^{LL}(\bar{S})(\sigma)(\{\vdash\}) \cup \alpha^{LL}(\bar{S})(\sigma)(X \fatsemi \longrightarrow)$ $\wr$lub preservation in Galois connections$\wr$

$= \quad \{\langle 0,\ \vdash\rangle\} \cup \alpha^{LL}(\bar{S})(\sigma)(X \fatsemi \longrightarrow)$
 $\wr$def. $\alpha^{LL}(\bar{S})(\sigma)$ with $i = 0$ so $\sigma_1 \ldots \sigma_i = \epsilon$ and $\{\vdash\}.\bar{S} \triangleq \{\vdash\}$$\wr$

We go on with the evaluation of $\alpha^{LL}(\bar{S})(\sigma)(X \fatsemi \longrightarrow)$

$= \quad \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi \xrightarrow{\ell'} \varpi' \mid \theta \xrightarrow{\ell} \varpi \in X \wedge \varpi \xrightarrow{\ell'} \varpi' \in \longrightarrow\}$ $\wr$def. $\fatsemi$ and $\longrightarrow$$\wr$

$= \quad \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \vdash \xrightarrow{(\!|A} \dashv[A \to \bullet\eta] \mid \theta \xrightarrow{\ell} \vdash \in X \wedge A \to \eta \in \mathcal{R}\} \cup$ (A)

$\{\theta \xrightarrow{\ell} \varpi[A \to \eta\bullet a\eta'] \xrightarrow{a} \varpi[A \to \eta a\bullet\eta'] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta\bullet a\eta'] \in X \wedge A \to \sigma a\sigma' \in \mathcal{R}\} \cup$ (B)

$\{\theta \xrightarrow{\ell} \varpi[A \to \eta\bullet B\eta'] \xrightarrow{(\!|B} \varpi[A \to \eta B\bullet\eta'][B \to \bullet\varsigma] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta\bullet B\eta'] \in X \wedge A \to \sigma B\sigma' \in \mathcal{R} \wedge B \to \varsigma \in \mathcal{R}\} \cup$ (C)

$$\{\theta \xrightarrow{\ell} \varpi[A \to \eta.] \xrightarrow{A\rangle\!\!\rangle} \varpi \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.] \in X \wedge A \to \eta \in \mathscr{R}\}) \qquad \text{(D)}$$

$\wr$by cases (1)–(4) of the def. of $\longrightarrow\wr$

$= \alpha^{LL}(\bar{S})(\sigma)(A) \cup \alpha^{LL}(\bar{S})(\sigma)(B) \cup \alpha^{LL}(\bar{S})(\sigma)(C) \cup \alpha^{LL}(\bar{S})(\sigma)(D)$

$\wr$lub preservation in Galois connections$\wr$

We now have four cases, as follows

— $\alpha^{LL}(\bar{S})(\sigma)(A)$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \vdash \xrightarrow{\langle\!\langle A} \dashv[A \to .\eta] \mid \theta \xrightarrow{\ell} \vdash \in X \wedge A \to \eta \in \mathscr{R}\})$ $\quad\wr$def. case (A)$\wr$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\vdash \xrightarrow{\langle\!\langle A} \dashv[A \to .\eta] \mid \vdash \in X \wedge A \to \eta \in \mathscr{R}\})$

$\wr X$ is an iterate of $\vec{\mathsf{F}}^{\partial}[\![\mathscr{G}]\!]$ so included in the prefix derivation semantics $\vec{\mathsf{S}}^{\partial}[\![\mathscr{G}]\!]$ hence, by Theorem 7, the only trace of the form $\theta \xrightarrow{\ell} \vdash$ is $\vdash\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 = \vdash \xrightarrow{\langle\!\langle \bar{S}} \dashv[\bar{S} \to .\varsigma] \wedge \vdash \in X \wedge \bar{S} \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_1\}$

$\wr$def. $\alpha^{LL}(\bar{S})(\sigma)$, selection $\bullet.\bar{S}$, and $\in\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 = \vdash \xrightarrow{\langle\!\langle \bar{S}} \dashv[\bar{S} \to .\varsigma] \wedge \vdash \in X \wedge \bar{S} \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \epsilon = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_1\}$

$\wr$def. $\alpha^{\tau}\wr$

$= \{\langle 0,\ \dashv[\bar{S} \to .\varsigma]\rangle \mid \vdash \in X \wedge \bar{S} \to \varsigma \in \mathscr{R}\}$ $\quad\wr$since $\epsilon = \sigma_1 \ldots \sigma_i \Longleftrightarrow i = 0\wr$

$= \{\langle 0,\ \dashv[\bar{S} \to .\varsigma]\rangle \mid \langle 0,\ \vdash\rangle \in \alpha^{LL}(\bar{S})(\sigma)(X) \wedge \bar{S} \to \varsigma \in \mathscr{R}\}$ $\quad\wr$def. $\alpha^{LL}\wr$

— $\alpha^{LL}(\bar{S})(\sigma)(B)$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.a\eta'] \xrightarrow{a} \varpi[A \to \eta a.\eta'] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.a\eta'] \in X \wedge A \to \sigma a\sigma' \in \mathscr{R}\})$ $\quad\wr$def. case (B)$\wr$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.a\eta'] \xrightarrow{a} \varpi[A \to \eta a.\eta'] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.a\eta'] \in X\})$

$\wr$because X is an iterate of $\vec{\mathsf{F}}^{\partial}[\![\mathscr{G}]\!]$ so, by Lemma 7, $[A \to \eta.a\eta']$ can be on the stack only if $A \to \sigma a\sigma'$ is a grammar rule in $\mathscr{R}\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.a\eta'] \xrightarrow{a} \varpi'[A \to \eta a.\eta'] \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.a\eta'] \in X.\bar{S}\} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m\}$ $\quad\wr$def. $\alpha^{LL}(\bar{S})(\sigma)$ and selection $\bullet.\bar{S}\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \in X.\bar{S}, \varpi_{m-1} = \varpi'[A \to \eta.a\eta'], \ell_{m-1} = a, \varpi_m = \varpi'[A \to \eta a.\eta'] : m \geqslant 1 \wedge i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta'' \xrightarrow{\ell_{m-1}} \varpi_m) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m\}$ $\quad\wr$def. $\in$ with $\theta = \theta'' \xrightarrow{\ell_{m-1}} \varpi_m\wr$

$= \{\langle i,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_m \in X.\bar{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta'')a = \sigma_1 \ldots \sigma_i\}$

$\wr$def. α^{τ} and setting the dummy variable m to $m - 1 \geqslant 0\wr$

$= \{\langle i,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_m \in X.\bar{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [1, |\sigma|] \wedge \alpha^{\tau}(\theta'')a = \sigma_1 \ldots \sigma_i\}$ $\wr$since $\alpha^{\tau}(\theta'')a = \sigma_1 \ldots \sigma_i$ implies $1 \leqslant i \leqslant |\sigma|\wr$

$= \{\langle i+1,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_m \in X.\bar{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [0, |\sigma| - 1] \wedge \alpha^{\tau}(\theta'')a = \sigma_1 \ldots \sigma_{i+1}\}$ $\quad\wr$setting the dummy variable i to $i + 1\wr$

$= \{\langle i+1,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_m \in X.\bar{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [0, |\sigma| - 1] \wedge \alpha^{\tau}(\theta'') = \sigma_1 \ldots \sigma_i \wedge \sigma_{i+1} = a\}$ $\quad\wr$def. equality of sequences$\wr$

$= \{\langle i+1,\ \varpi[A \to \eta a.\eta']\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.a\eta'] = \varpi_m \wedge a = \sigma_{i+1}\}$ $\quad\wr$since $\sigma_{i+1} = a$ implies $i + 1 \leqslant |\sigma|\wr$

$= \{\langle i+1,\ \varpi[A \to \eta a.\eta']\rangle \mid \langle i,\ \varpi[A \to \eta.a\eta']\rangle \in \alpha^{LL}(\bar{S})(\sigma)(X) \wedge a = \sigma_{i+1}\}$ $\quad\wr$def. $\in$ and $\alpha^{LL}(\bar{S})(\sigma)\wr$

— $\alpha^{LL}(\bar{S})(\sigma)(C)$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \xrightarrow{\langle\!\langle B} \varpi[A \to \eta B.\eta'][B \to .\varsigma] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \in X \wedge A \to \sigma B\sigma' \in \mathscr{R} \wedge B \to \varsigma \in \mathscr{R}\})$

$\wr$def. case (C)$\wr$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \xrightarrow{\langle\!\langle B} \varpi[A \to \eta B.\eta'][B \to .\varsigma] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \in X \wedge B \to \varsigma \in \mathscr{R}\})$

$\wr$because X is an iterate of $\vec{\mathsf{F}}^{\partial}[\![\mathscr{G}]\!]$ so by Lemma 7, $[A \to \eta.B\eta']$ can be on the stack only if $A \to \sigma B\sigma'$ is a grammar rule in $\mathscr{R}\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.B\eta'] \xrightarrow{\langle\!\langle B} \varpi'[A \to \eta B.\eta'][B \to .\varsigma] \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.B\eta'] \in X.\bar{S} \wedge B \to \varsigma \in \mathscr{R}\} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m\}$ $\quad\wr$def. $\alpha^{LL}(\bar{S})(\sigma)$ and selection $\bullet.\bar{S}\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \in X.\bar{S}, \varpi_{m-1} = \varpi'[A \to \eta.B\eta'], \ell_{m-1} = (\!|B, \varpi_m = \varpi'[A \to \eta B.\eta'][B \to .\varsigma] : m \geqslant 1 \wedge B \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'' \xrightarrow{\ell_{m-1}} \varpi_m) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m\}$ ⦃def. $\in$ and $\theta = \theta'' \xrightarrow{\ell_{m-1}} \varpi_m$⦄

$= \{\langle i,\ \varpi'[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \in X.\bar{S}, \varpi_{m-1} = \varpi'[A \to \eta.B\eta'] : m \geqslant 1 \wedge B \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'') = \sigma_1 \ldots \sigma_i\}$ ⦃def. α^τ⦄

$= \{\langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.B\eta'] = \varpi_m \wedge B \to \varsigma \in \mathscr{R}\}$ ⦃setting the dummy variable m to $m-1 \geqslant 0$ and $\theta = \theta''$⦄

$= \{\langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \langle i,\ \varpi[A \to \eta.B\eta']\rangle \in \alpha^{LL}(\bar{S})(\sigma)(X) \wedge B \to \varsigma \in \mathscr{R}\}$ ⦃def. $\in$ and $\alpha^{LL}(\bar{S})(\sigma)$⦄

— $\alpha^{LL}(\bar{S})(\sigma)(D)$

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.] \xrightarrow{A|\!)} \varpi \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.] \in X \wedge A \to \eta \in \mathscr{R}\})$ ⦃def. case (D)⦄

$= \alpha^{LL}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.] \xrightarrow{A|\!)} \varpi \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.] \in X\})$

⦃because X is an iterate of $\vec{\mathsf{F}^{\partial}}[\![\mathcal{G}]\!]$ so, by Lemma 7, $[A \to \eta.]$ can be on the stack only if $A \to \eta$ is a grammar rule in $\mathscr{R}$⦄

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \in X.\bar{S}\} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m\}$ ⦃def. $\alpha^{LL}(\bar{S})(\sigma)$ & $\bullet.\bar{S}$⦄

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \xrightarrow{A|\!)} \varpi' \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \in X.\bar{S}\} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'') = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.] = \varpi_{m-1}\}$

⦃setting $\theta = \theta'' \xrightarrow{\ell_{m-1}} \varpi_m$ with $\ell_{m-1} = A|\!)$, $\varpi_m = \varpi'$ and $\varpi_{m-1} = \varpi[A \to \eta.]$ since $\alpha^\tau(\theta) = \alpha^\tau(\theta'')$⦄

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.] = \varpi_m\}$

⦃def. $\in$ & setting dummy variable m to $m-1 \geqslant 0$⦄

$= \{\langle i,\ \varpi\rangle \mid \langle i,\ \varpi[A \to \eta.]\rangle \in \alpha^{LL}(\bar{S})(\sigma)(X)\}$ ⦃def. $\in$ and $\alpha^{LL}(\bar{S})(\sigma)$⦄. □

23.2. The nonrecursive predictive parsing algorithm

Observe that, by Example 107, $\mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ is exactly the set of reachable states of the transition system $\langle [0, |\sigma|] \times \mathcal{S}, \xrightarrow{\text{LL}}\rangle$ where

$$\langle 0,\ \vdash\rangle \xrightarrow{\text{LL}} \langle 0,\ \dashv[\bar{S} \to .\eta]\rangle \qquad \bar{S} \to \eta \in \mathscr{R} \tag{56}$$

$$\langle i,\ \varpi[A \to \eta.\sigma_{i+1}\eta']\rangle \xrightarrow{\text{LL}} \langle i+1,\ \varpi[A \to \eta\sigma_{i+1}.\eta']\rangle \tag{57}$$

$$\langle i,\ \varpi[A \to \eta.B\eta']\rangle \xrightarrow{\text{LL}} \langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \quad B \to \varsigma \in \mathscr{R} \tag{58}$$

$$\langle i,\ \varpi[A \to \eta.]\rangle \xrightarrow{\text{LL}} \langle i,\ \varpi\rangle \tag{59}$$

with initial state $\langle 0,\ \vdash\rangle$. By Theorem 88, parsing is therefore reduced to proving that the final state $\langle |\sigma|,\ \dashv\rangle$ is reachable (which can be done by computing the iterates of $\mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ or equivalently by exploring the descendants of the transition relation $\xrightarrow{\text{LL}}$ with backtracking when reaching a dead end [4, Alg. 4.1, Section 4.1.3]).

Example 90. Consider the grammar $\mathcal{G} = \langle\{a, b\},\ \{A\},\ A,\ \{A \to A, A \to a\}\rangle$. For the input sentence $\sigma = a$ we have

$\langle 0,\ \vdash\rangle \xrightarrow{\text{LL}} \langle 0,\ \dashv[A \to .a]\rangle$ ⦃from initial state by (56) with rule $A \to a$⦄

$\langle 1,\ \dashv[A \to a.]\rangle \xrightarrow{\text{LL}} \langle 1,\ \dashv\rangle$ ⦃by (57) since $\sigma_1 = a$ and (59), which is a final state⦄.

On the other hand, the transitions for $\sigma = b$ either lead to dead ends or do not terminate

$\langle 0,\ \vdash\rangle \xrightarrow{\text{LL}} \langle 0,\ \dashv[A \to .A]\rangle$

⦃from initial state by (56) with rule $A \to A$ since $A \to a$ would lead to a dead end because $\sigma_1 = b \neq a$⦄

$\xrightarrow{\text{LL}}$ $\langle 0,\ \dashv[A \to .A][A \to .A]\rangle$ ⦃by (58) with rule $A \to A$ since $A \to a$ would lead to a dead end because $\sigma_1 = b \neq a$⦄

$\xrightarrow{\text{LL}}$ $\langle 0,\ \dashv[A \to .A][A \to .A][A \to .A]\rangle$ ⦃by (58) with rule $A \to A$ since $A \to a$ would lead to a dead end because $\sigma_1 = b \neq a$⦄

$\xrightarrow{\text{LL}}$ $\ldots$ ⦃etc, ad infinitum, without any possibility of success or failure in a blocking state⦄. □

Theorem 91. *The nonrecursive predictive parsing algorithm for a grammar $\mathcal{G} = \langle \mathcal{T}, \mathcal{N}, \overline{S}, \mathcal{R}\rangle$ terminates (i.e. the transition relation $\xrightarrow{\text{LL}}$ has no infinite trace for all input sentences $\sigma \in \mathcal{T}^\star$) if and only if the grammar $\mathcal{G}$ has no* left recursion *(that is $\exists A \in \mathcal{N} : \exists \eta \in \mathcal{V}^\star : A \stackrel{+}{\Longmapsto}_{\mathcal{G}} A\eta$).*

Proof. By reductio ad absurdum, assume that there exists an infinite trace for some input σ.

Because (56) is only applicable in the initial state $\langle 0, \vdash\rangle$ and (57) strictly increases i which is bounded by the finite length $|\sigma|$ of the input sentence σ, there must be a point in the infinite trace where only (58) and (59) are applicable and the stack has minimal height (no stack appearing later in the trace can have a strictly less height).

This stack cannot be reduced to $\dashv$ since in this case the state would be final or blocking. So the corresponding state has necessarily the form $\langle i, \varpi[A_1 \to \eta_1 \centerdot A_2\eta_1']\rangle$ (the stack cannot be of the form $\varpi[A \to \eta\centerdot]$ since then (59) would strictly reduce the height of the stack nor of the form $\varpi[A \to \eta\centerdot a\eta']$ which would be a dead end since (57) is no longer applicable). All later state in the trace correspond to the position i since (57) is assumed to be no longer applicable in the trace. Moreover no later state in the trace can be of the form $\langle i, \varpi[A_1 \to \eta_1 A_2\eta_1'\centerdot]\rangle$ since (59) would then strictly reduce the height of the stack, which would be in contradiction with the minimality of the height of the stack from now on. So there is a later position in the trace of this form with η_1' of minimal length.

Assume, by induction hypothesis, that the trace contains a later state of the form $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k\centerdot A_{k+1}\eta_k']\rangle$ with η_k' of minimal length (no later state can be of the form $\langle i, \varpi[A_1 \to \eta_1 A_2\eta_1'\centerdot] \ldots [A_k \to \eta_k\centerdot A_{k+1}'\eta_k'']\rangle$ with $|\eta_k''| < \eta_k'$).

The next state is then $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \centerdot\eta]\rangle$ by (58). All later states have necessarily the form $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \eta_{k+1}\centerdot A_{k+2}\eta_{k+1}']\rangle$ with $\eta_{k+1} \Longmapsto_{\mathcal{G}} \epsilon$ and $\eta_{k+1}' \neq \epsilon$.

- We have $\eta_{k+1}' \neq \epsilon$ since the stack cannot be of the form $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \eta\centerdot]\rangle$ since then (59) would strictly reduce the height of the stack in contradiction with the minimality of η_k' nor of the form $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \eta_{k+1}\centerdot a\eta_{k+1}']\rangle$ which would be a dead end since (57) is no longer applicable).
- It follows that the only applicable transitions to reach $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \eta_{k+1}\centerdot A_{k+2}\eta_{k+1}']\rangle$ from $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \eta_{k+1} A_{k+2}\eta_{k+1}'\centerdot]\rangle$ are (58) with $B \to \varsigma \in \mathcal{R}$ immediately followed by (59) so that $\varsigma = \epsilon$ proving that $\eta_{k+1} \Longmapsto_{\mathcal{G}} \epsilon$.

So there is one later state of the form $\langle i, \varpi[A_1 \to \eta_1 A_2\centerdot\eta_1'] \ldots [A_k \to \eta_k A_{k+1}\centerdot\eta_k'][A_{k+1} \to \eta_{k+1}\centerdot A_{k+2}\eta_{k+1}']\rangle$ with η_{k+1}' of minimal length. This means that this construction can go on for ever.

Since there are only finitely many grammar rules, some rule, say $A_1 \to \eta_1 A_2\eta_1'$, must be applied at least twice. So we have a finite sequence of grammar rules $A_1 \to \eta_1 A_2\eta_1', \ldots, A_k \to \eta_k A_{k+1}\eta_k'$, $k \geqslant 1$, where we have shown that $\eta_1 \Longmapsto_{\mathcal{G}} \epsilon, \ldots, \eta_k \Longmapsto_{\mathcal{G}} \epsilon$ and $A_{k+1} = A_1$. It follows that we have a left recursion for A_1 since by def. (47) of $\Longmapsto_{\mathcal{G}}$, we have $A_1 \stackrel{+}{\Longmapsto}_{\mathcal{G}} A_{k+1}\eta_k' \ldots \eta_1' = A_1\eta_k' \ldots \eta_1'$.

Because there are finitely many nonterminals, terminals and grammar rules, the transition system has a finitely bounded nondeterminism. So if all traces are finite, there are finitely many of them, whence the iterates in the iterative computation of $\mathbf{lfp}^{\subseteq}\, \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ do converge in finitely many steps. $\square$

23.3. Nonrecursive predictive parsing with lookahead

The nondeterminism in predictive parsing can be reduced by driving the right context in derivations (as approximated using First and Follow).

23.4. Right context in derivations

We start by elucidating the rôle of the right context in derivations.

Given a stack $\varpi = \dashv[A_1 \to \eta_1\centerdot\eta_1'] \ldots [A_p \to \eta_p\centerdot\eta_p']$, $p \geqslant 0$ where $\varpi = \dashv$ when $p = 0$, we define the *right context* $\varpi^\vartriangle$ *of* ϖ as

$$\varpi^\vartriangle \quad \stackrel{\Delta}{=} \quad \eta_p'\eta_{p-1}' \ldots \eta_2'\eta_1'$$

with $\eta_p'\eta_{p-1}' \ldots \eta_2'\eta_1' = \epsilon$ when $p = 0$.

Theorem 92. *Let $\varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{i-1} \xrightarrow{\ell_{i-1}} \varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n \in \mathsf{S}^{\vec{d}}[\![\mathcal{G}]\!]$ be a maximal derivation of the grammar $\mathcal{G} = \langle \mathcal{T}, \mathcal{N}, \overline{S}, \mathcal{R}\rangle$ with $i > 0$. Then*

$$\varpi_i^\vartriangle \quad \stackrel{\star}{\Longmapsto}_{\mathcal{G}} \quad \alpha^\tau(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n). \quad \square$$

We call $\alpha^\tau(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)$ the *terminal right context* of ϖ_i.

Proof. The facts that $n > 1$, $\varpi_0 = \vdash$ and $\varpi_n = \dashv$ follow from Lemma 9. By Lemma 7, the stack ϖ_i has the shape $\varpi_i = \dashv[A_1 \to \eta_1 \bullet \eta'_1] \dots [A_p \to \eta_p \bullet \eta'_p]$, $p \geqslant 0$ when $i > 0$. The proof is by induction on the length of the suffix $\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n$.

— If $i = n$ then $\varpi_i = \varpi_n = \dashv$ so $\varpi_i{}^{\triangle} = \eta'_p\eta'_{p-1}\dots\eta'_2\eta'_1 = \epsilon \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon = \alpha^{\tau}(\dashv) = \alpha^{\tau}(\varpi_n) = \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)$ by def. α^{τ} and $i = n$.

— Otherwise, for the induction step, $i < n$. By def. (5) of the transition-based maximal derivation semantics $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!]$, $\varpi_i \xrightarrow{\ell_i} \varpi_{i+1}$ is a transition of the labelled transition system $\langle \mathcal{S},\ \mathcal{L},\ \longrightarrow,\ \vdash\rangle$. We go on by cases.

— The case (1) of a transition $\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} = \vdash \xrightarrow{(\!|A} \dashv[A \to \bullet\eta]$ is impossible since $i > 0$ so ϖ_i is not the initial state $\vdash$

— In case (2) $\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} = \varpi[A \to \eta\bullet a\eta'] \xrightarrow{a} \varpi[A \to \eta a\bullet\eta']$, we have $\varpi_i{}^{\triangle}$

$$
\begin{array}{lll}
= & a\varpi_{i+1}{}^{\triangle} & \\
& \multicolumn{2}{l}{\wr\text{since } \varpi_i = \varpi[A \to \eta\bullet a\eta'],\ \text{def. } (\varpi[A \to \eta\bullet a\eta'])^{\triangle},\ \text{def. } (\varpi[A \to \eta a\bullet\eta'])^{\triangle} \text{ and } \varpi_{i+1} = \varpi[A \to \eta a\bullet\eta']\wr} \\
\overset{\star}{\Longmapsto}_{\mathcal{G}} & a\alpha^{\tau}(\varpi_{i+1} \xrightarrow{\ell_{i+1}} \varpi_{i+2} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr\text{ind. hyp.}\wr \\
= & \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr\text{def. } \alpha^{\tau} \text{ since } \ell_i = a \in \mathcal{T}\wr
\end{array}
$$

— In case (3) $\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} = \varpi[A \to \eta\bullet B\eta'] \xrightarrow{(\!|B} \varpi[A \to \eta B\bullet\eta'][B \to \bullet\varsigma]$, we have

$$
\begin{array}{lll}
& \varpi_i{}^{\triangle} \ = \ B\sigma'\varpi^{\triangle} & \wr\text{def. } \varpi_i{}^{\triangle} = (\varpi[A \to \eta\bullet B\eta'])^{\triangle}\wr \\
\Longmapsto_{\mathcal{G}} & \varsigma\sigma'\varpi^{\triangle} & \wr\text{def. (47) of } \Longmapsto_{\mathcal{G}} \text{ since } B \to \varsigma \in \mathcal{R} \text{ by def. (3) of } \xrightarrow{(\!|B}\wr \\
= & \varpi_{i+1}{}^{\triangle} & \wr\text{def. } \varpi_{i+1}{}^{\triangle} = \varpi[A \to \eta B\bullet\eta'][B \to \bullet\varsigma]\wr \\
\overset{\star}{\Longmapsto}_{\mathcal{G}} & \alpha^{\tau}(\varpi_{i+1} \xrightarrow{\ell_{i+1}} \varpi_{i+2} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr\text{ind. hyp.}\wr \\
= & \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr\text{def. } \alpha^{\tau} \text{ since } \ell_i = (\!|B\wr
\end{array}
$$

and so $\varpi_i{}^{\triangle} \overset{\star}{\Longmapsto}_{\mathcal{G}} \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)$ by def. of $\overset{\star}{\Longmapsto}_{\mathcal{G}}$ as the reflexive transitive closure of $\Longmapsto_{\mathcal{G}}$

— Finally, in case (4) $\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} = \varpi[A \to \eta\bullet] \xrightarrow{A|\!)} \varpi$, we have $\varpi_i{}^{\triangle}$

$$
\begin{array}{lll}
= & \varpi_{i+1}{}^{\triangle} & \wr\text{since } \varpi_i = \varpi[A \to \eta\bullet],\ \text{def. } (\varpi[A \to \eta\bullet])^{\triangle}, \text{ and } \varpi_{i+1} = \varpi\wr \\
\overset{\star}{\Longmapsto}_{\mathcal{G}} & \alpha^{\tau}(\varpi_{i+1} \xrightarrow{\ell_{i+1}} \varpi_{i+2} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr\text{ind. hyp.}\wr \\
= & \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr\text{def. } \alpha^{\tau} \text{ since } \ell_i = A|\!)\wr. \quad \square
\end{array}
$$

23.5. First approximation of the right context in derivations

In order to approximate the right contexts in derivations by their first symbol, we define

$$
\begin{array}{rcl}
\vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta\bullet\eta'] & \overset{\triangle}{=} & \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \oplus^{1} \mathsf{S}^{f}[\![\mathcal{G}]\!](A) \\
= & & (\!|\mathsf{S}^{f}[\![\mathcal{G}]\!](A) \neq \varnothing \mathrel{?} (\vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \setminus \{\epsilon\}) \cup (\!|\epsilon \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \mathrel{?} \mathsf{S}^{f}[\![\mathcal{G}]\!](A) \mathrel{\wr} \varnothing|\!) \mathrel{\wr} \varnothing|\!) \\
= & & (\!|\mathsf{S}^{f}[\![\mathcal{G}]\!](A) \neq \varnothing \mathrel{?} (\vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \setminus \{\epsilon\}) \cup (\!|\vec{\mathsf{S}}^{\epsilon}[\![\mathcal{G}]\!](\eta') \mathrel{?} \mathsf{S}^{f}[\![\mathcal{G}]\!](A) \mathrel{\wr} \varnothing|\!) \mathrel{\wr} \varnothing|\!).
\end{array}
\tag{60}
$$

Corollary 93. *Let $\varpi_0 \xrightarrow{\ell_0} \varpi_1 \dots \varpi_{i-1} \xrightarrow{\ell_{i-1}} \varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n \in \mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].\bar{S}$, $i > 0$ be a maximal derivation of the grammar $\mathcal{G} = \langle \mathcal{T},\ \mathcal{N},\ \bar{S},\ \mathcal{R}\rangle$ from the grammar start symbol $\bar{S}$. Then*

$$\alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \dots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)\dashv \quad = \quad a\sigma$$

where $\varpi_i = \varpi'_i[A \to \eta\bullet\eta']$, $a \in \mathcal{T} \cup \{\dashv\}$, $\sigma \in (\mathcal{T} \cup \{\dashv\})^{\star}$ and

$$a \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta\bullet\eta']. \quad \square$$

Proof. By Lemma 7 and $i > 0$, we have ϖ_i of the form $\varpi_i = \dashv[A_1 \to \eta_1 A_2\bullet\eta'_1][A_2 \to \eta_2 A_3\bullet\eta'_2] \dots [A_n \to \eta_n\bullet\eta'_n] = \varpi'_i[A \to \eta\bullet\eta']$ where $\varpi'_i = \dashv[A_1 \to \eta_1 A_2\bullet\eta'_1][A_2 \to \eta_2 A_3\bullet\eta'_2] \dots [A_{n-1} \to \eta_{n-1} A_n\bullet\eta'_{n-1}]$, $A_n = A$, $\eta_n = \eta$ and $\eta'_n = \eta'$.

Since the trace belongs to $\mathsf{S}^{\hat{d}}[\![\mathcal{G}]\!].\overline{S}$, the definition of the selection $\bullet.\overline{S}$ and Lemma 7 imply that $A_1 = \overline{S}$ so $\varpi_i = \dashv[\overline{S} \to \eta_1 A_2 \bullet \eta'_1][A_2 \to \eta_2 A_3 \bullet \eta'_2] \ldots [A_n \to \eta_n \bullet \eta'_n]$ where, again by Lemma 7, $\overline{S} \to \eta_1 A_2 \eta'_1 \in \mathscr{R}, A_2 \to \eta_2 A_3 \eta'_2 \in \mathscr{R}, \ldots A_n \to \eta_n \eta'_n = A \to \eta\eta' \in \mathscr{R}$ are all grammar rules.

It follows, by induction on n and def. (47) of $\Longmapsto_{\mathcal{G}}$, that $\overline{S} \Longmapsto_{\mathcal{G}} \eta_1 A_2 \eta'_1 \Longmapsto_{\mathcal{G}} \eta_1\eta_2 A_3 \eta'_2 \eta'_1 \ldots \Longmapsto_{\mathcal{G}} \eta_1\eta_2 \ldots \eta_{n-1} A_n \eta'_{n-1} \ldots \eta'_2\eta'_1 = \eta_1\eta_2 \ldots \eta_{n-1} A \eta'_{n-1} \ldots \eta'_2\eta'_1 \Longmapsto_{\mathcal{G}} \eta_1\eta_2 \ldots \eta_{n-1}\eta\eta'\eta'_{n-1} \ldots \eta'_2\eta'_1$ proving that

$$\begin{array}{rrcl} & \overline{S} & \overset{\star}{\Longmapsto}_{\mathcal{G}} & \eta_1\eta_2 \ldots \eta_{n-1} A \eta'_{n-1} \ldots \eta'_2\eta'_1 \\ \text{and} & \overline{S} & \overset{\star}{\Longmapsto}_{\mathcal{G}} & \eta_1\eta_2 \ldots \eta_{n-1}\eta\eta'\eta'_{n-1} \ldots \eta'_2\eta'_1. \end{array}$$

We have $\eta'(\varpi'_i)^{\triangle}$

$$\begin{array}{ll} = \quad \varpi_i{}^{\triangle} & \wr \text{def. } \bullet^{\triangle} \text{ and since } \varpi_i = \varpi'_i[A \to \eta \bullet \eta'] \text{, as shown above.} \wr \\ \overset{\star}{\Longmapsto}_{\mathcal{G}} \quad \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) & \wr \text{by Theorem 92} \wr \end{array}$$

$\alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_i \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)\dashv \in (\mathscr{T} \cup \{\dashv\})^+$ is not empty whence of the form $a\sigma$ where $a \in \mathscr{T} \cup \{\dashv\}$ and $\sigma \in (\mathscr{T} \cup \{\dashv\})^\star$. We have

$$\begin{array}{ll} \quad a \in \{a\} = \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](a\sigma) & \wr \text{by def. } \in \text{ and Theorem 74} \wr \\ = \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)\dashv) & \wr \text{since } a\sigma = \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_i \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n)\dashv \wr \\ = \{a \in \mathscr{T} \mid \exists \sigma \in \mathscr{T}^\star : \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\dashv \mid \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\} & \\ & \wr \text{def. (51) of the extension of } \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!] \text{ to } \mathscr{V}^\star\{\dashv\} \mapsto \wp(\mathscr{T} \cup \{\dashv\}) \wr \\ \subseteq \{a \in \mathscr{T} \mid \exists \sigma \in \mathscr{T}^\star : \eta'(\varpi'_i)^{\triangle} \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\dashv \mid \eta'(\varpi'_i)^{\triangle} \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\} & \\ \qquad \wr \text{since } \eta'(\varpi'_i)^{\triangle} \overset{\star}{\Longmapsto}_{\mathcal{G}} \alpha^{\tau}(\varpi_i \xrightarrow{\ell_i} \varpi_{i+1} \ldots \varpi_{n-1} \xrightarrow{\ell_{n-1}} \varpi_n) \text{ and } \overset{\star}{\Longmapsto}_{\mathcal{G}} \text{ is transitive} \wr & \\ = \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta'(\varpi'_i)^{\triangle}\dashv) & \wr \text{def. (51) of the extension of } \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!] \text{ to } \mathscr{V}^\star\{\dashv\} \mapsto \wp(\mathscr{T} \cup \{\dashv\}) \wr \\ = \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \oplus^1 \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!]((\varpi'_i)^{\triangle}\dashv) & \wr \text{by (50)} \wr. \end{array}$$

Moreover $\vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!]((\varpi'_i)^{\triangle}\dashv)$

$$\begin{array}{lll} = & \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!]((\dashv[A_1 \to \eta_1 A_2 \bullet \eta'_1][A_2 \to \eta_2 A_3 \bullet \eta'_2] \ldots [A_{n-1} \to \eta_{n-1} A_n \bullet \eta'_{n-1}])^{\triangle}\dashv) & \\ & & \wr \text{since } \varpi'_i = \dashv[A_1 \to \eta_1 A_2 \bullet \eta'_1][A_2 \to \eta_2 A_3 \bullet \eta'_2] \ldots [A_{n-1} \to \eta_{n-1} A_n \bullet \eta'_{n-1}] \wr \\ = & \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta'_1\eta'_2 \ldots \eta'_{n-1}\dashv) & \wr \text{def. } \bullet^{\triangle} \wr \\ = & \{a \in \mathscr{T} \mid \exists \sigma \in \mathscr{T}^\star : \eta'_1\eta'_2 \ldots \eta'_{n-1} \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\dashv \mid \eta'_1\eta'_2 \ldots \eta'_{n-1} \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\} & \\ & & \wr \text{def. (51) of the extension of } \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!] \text{ to } \mathscr{V}^\star\{\dashv\} \mapsto \wp(\mathscr{T} \cup \{\dashv\}) \wr \\ = & \{a \in \mathscr{T} \mid \exists \sigma \in \mathscr{T}^\star : \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta_1\eta_2 \ldots \eta_{n-1} A \eta'_{n-1} \ldots \eta'_2\eta'_1 \wedge \eta'_1\eta'_2 \ldots \eta'_{n-1} \overset{\star}{\Longmapsto}_{\mathcal{G}} a\sigma\} \cup \{\dashv \mid \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} & \\ & \eta_1\eta_2 \ldots \eta_{n-1} A \eta'_{n-1} \ldots \eta'_2\eta'_1 \wedge \eta'_1\eta'_2 \ldots \eta'_{n-1} \overset{\star}{\Longmapsto}_{\mathcal{G}} \epsilon\} \quad \wr \text{since } \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta_1\eta_2 \ldots \eta_{n-1} A \eta'_{n-1} \ldots \eta'_2\eta'_1 \text{, as shown above} \wr & \\ \subseteq & \{a \in \mathscr{T} \mid \exists \sigma \in \mathscr{T}^\star : \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta_1\eta_2 \ldots \eta_{n-1} A a \sigma\} \cup \{\dashv \mid \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta_1\eta_2 \ldots \eta_{n-1} A\} & \\ & \quad \wr \text{def. } \overset{\star}{\Longmapsto}_{\mathcal{G}} \text{ and } \Longmapsto_{\mathcal{G}} \text{ so that } \eta \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta'\eta'' \text{ and } \eta'' \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta''' \text{ implies } \eta \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta'\eta''' \text{ and } \epsilon \text{ neutral element of concatenation} \wr & \\ \subseteq & \{a \in \mathscr{T} \mid \exists \eta, \eta' : \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta A a \eta'\} \cup \{\dashv \mid \exists \eta : \overline{S} \overset{\star}{\Longmapsto}_{\mathcal{G}} \eta A\} & \wr \text{def. } \exists \wr \\ = & \mathsf{S}^{f}[\![\mathcal{G}]\!](A) & \wr \text{by Theorem 79} \wr. \end{array}$$

By def. $\oplus^1$, we conclude that $a \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \oplus^1 \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!]((\varpi'_i)^{\triangle}\dashv) \subseteq \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!](\eta') \oplus^1 \mathsf{S}^{f}[\![\mathcal{G}]\!](A) \triangleq \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta \bullet \eta']$. □

If the input sentence σ derives from the start symbol $\overline{S}$ then the right context $\varpi^{\triangle}$ of the stack ϖ in $\langle i,\ \varpi\rangle$ should derive in the rest $\sigma_{i+1} \ldots \sigma_n$ of the input sentence. In order to introduce a lookahead, this can be approximated by the fact that, according to Corollary 93, the first symbol of this right context should be σ_{i+1} (which, by definition, is $\dashv$ when $i = n$ so that $\sigma_{|\sigma|+1} \triangleq \dashv$).

$$\begin{array}{c} \alpha^{LL(1)} \triangleq \boldsymbol{\lambda}\, \overline{S} \bullet \boldsymbol{\lambda}\, \sigma \bullet \boldsymbol{\lambda}\, X \bullet \{\langle i,\ \varpi\rangle \mid \exists \theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\overline{S} : \\ i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m \wedge \forall \varpi' \in \mathscr{S}, A \to \eta\eta' \in \mathscr{R} : \\ (\varpi = \varpi'[A \to \eta \bullet \eta'] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta \bullet \eta'])\}. \end{array}$$

The correctness of the nonrecursive predictive parser with lookahead is established by the following

Theorem 94. $\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S}) \Longleftrightarrow \langle|\sigma|,\ \dashv\rangle \in \alpha^{LL(1)}(\bar{S})(\sigma)(\vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!])$.

Proof. $\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S})$

$\Longleftrightarrow\ \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!].\bar{S} : |\sigma| \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_{|\sigma|} \wedge \dashv = \varpi_m$

≀as shown in the proof of Theorem 88≀

$\Longleftrightarrow\ \langle|\sigma|,\ \dashv\rangle \in \{\langle i,\ \varpi'\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!].\bar{S} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi' = \varpi_m \wedge \forall \varpi' \in \mathcal{S}, A \to \eta\eta' \in \mathcal{R} : (\varpi = \varpi'[A \to \eta\bullet\eta'] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta\bullet\eta'])\}$

≀def. $\in$ and $\forall \varpi' \in \mathcal{S} : \forall A \to \eta\eta' \in \mathcal{R} : \dashv \neq \varpi'[A \to \eta\bullet\eta']$≀

$\Longleftrightarrow\ \langle|\sigma|,\ \dashv\rangle \in \alpha^{LL(1)}(\bar{S})(\sigma)(\vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!])$ ≀def. $\alpha^{LL(1)}(\bar{S})(\sigma)$≀. □

The nonrecursive predictive parser with lookahead is obtained by expressing the abstract semantics in fixpoint form

Theorem 95. $\alpha^{LL(1)}(\bar{S})(\sigma)(\vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!]) = \boldsymbol{lfp}^{\subseteq}\, \mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma)$ *where* $\mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma) \in \wp([0, |\sigma|] \times \mathcal{S}) \mapsto \wp([0, |\sigma|] \times \mathcal{S})$ *is*

$$\begin{aligned}\mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma) = \boldsymbol{\lambda} X \bullet &\{\langle 0,\ \vdash\rangle\} \\ &\cup \{\langle 0,\ \dashv[\bar{S} \to \bullet\eta]\rangle \mid \langle 0,\ \vdash\rangle \in X \wedge \bar{S} \to \eta \in \mathcal{R} \wedge \sigma_1 \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][\bar{S} \to \bullet\eta]\} \\ &\cup \{\langle i+1,\ \varpi[A \to \eta a\bullet\eta']\rangle \mid \langle i,\ \varpi[A \to \eta\bullet a\eta']\rangle \in X \wedge \\ &\qquad\qquad a = \sigma_{i+1} \wedge \sigma_{i+2} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta a\bullet\eta']\} \\ &\cup \{\langle i,\ \varpi[A \to \eta B\bullet\eta'][B \to \bullet\varsigma]\rangle \mid \langle i,\ \varpi[A \to \eta\bullet B\eta']\rangle \in X \wedge \\ &\qquad\qquad B \to \varsigma \in \mathcal{R} \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][B \to \bullet\varsigma]\} \\ &\cup \{\langle i,\ \varpi\rangle \mid \langle i,\ \varpi[A \to \eta\bullet]\rangle \in X\}.\quad \square\end{aligned} \tag{61}$$

Proof. The proof is similar to that of Theorem 89. We have $\alpha^{LL(1)}(\bar{S})(\sigma)(\{\vdash\}) = \{\langle 0,\ \vdash\rangle\}$ by def. $\alpha^{LL(1)}(\bar{S})(\sigma)$ with $i = 0$ so $\sigma_1 \ldots \sigma_i = \epsilon$ and $\{\vdash\}.\bar{S} \triangleq \{\vdash\}$. We go on with the evaluation of $\alpha^{LL(1)}(\bar{S})(\sigma)(X \,{}_{\S}\, \longrightarrow) = \alpha^{LL(1)}(\bar{S})(\sigma)(A) \cup \alpha^{LL(1)}(\bar{S})(\sigma)(B) \cup \alpha^{LL(1)}(\bar{S})(\sigma)(C) \cup \alpha^{LL(1)}(\bar{S})(\sigma)(D)$ as in the proof of Theorem 89. We now have four cases, as follows

— $\alpha^{LL(1)}(\bar{S})(\sigma)(A)$

$= \alpha^{LL(1)}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \vdash \xrightarrow{(\!|A} \dashv[A \to \bullet\eta] \mid \theta \xrightarrow{\ell} \vdash \in X \wedge A \to \eta \in \mathcal{R}\})$ ≀def. case (A)≀

$= \alpha^{LL(1)}(\bar{S})(\sigma)(\{\vdash \xrightarrow{(\!|A} \dashv[A \to \bullet\eta] \mid \vdash \in X \wedge A \to \eta \in \mathcal{R}\})$

≀X is an iterate of $\vec{\mathsf{F}}^{\partial}[\![\mathcal{G}]\!]$ so included in the prefix derivation semantics $\vec{\mathsf{S}}^{\partial}[\![\mathcal{G}]\!]$ hence, by Theorem 7, the only trace of the form $\theta \xrightarrow{\ell} \vdash$ is $\vdash$≀

$\{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\vdash \xrightarrow{(\!|\bar{S}} \dashv[\bar{S} \to \bullet\varsigma] \mid \vdash \in X \wedge \bar{S} \to \varsigma \in \mathcal{R}\} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m \wedge \forall \varpi' \in \mathcal{S}, A \to \eta\eta' \in \mathcal{R} : (\varpi = \varpi'[A \to \eta\bullet\eta'] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta\bullet\eta'])\}$ ≀def. $\alpha^{LL(1)}(\bar{S})(\sigma)$ & $\bullet.\bar{S}$≀

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 = \vdash \xrightarrow{(\!|\bar{S}} \dashv[\bar{S} \to \bullet\varsigma] \wedge \vdash \in X \wedge \bar{S} \to \varsigma \in \mathcal{R} \wedge i \in [0, |\sigma|] \wedge \epsilon = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_1 \wedge \forall \varpi' \in \mathcal{S}, A \to \eta\eta' \in \mathcal{R} : (\varpi = \varpi'[A \to \eta\bullet\eta'] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta\bullet\eta'])\}$ ≀def. $\in$ and α^{τ}≀

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 = \vdash \xrightarrow{(\!|\bar{S}} \dashv[\bar{S} \to \bullet\varsigma] \wedge \vdash \in X \wedge \bar{S} \to \varsigma \in \mathcal{R} \wedge i \in [0, |\sigma|] \wedge \epsilon = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_1 \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][\bar{S} \to \bullet\varsigma]\}$ ≀since $\varpi = \varpi_1 = \dashv[\bar{S} \to \bullet\varsigma] = \varpi'[A \to \eta\bullet\eta']$ so $\varpi' = \dashv, A = \bar{S}, \eta = \epsilon$ and $\eta' = \varsigma$≀

$= \{\langle 0,\ \dashv[\bar{S} \to \bullet\varsigma]\rangle \mid \vdash \in X \wedge \bar{S} \to \varsigma \in \mathcal{R} \wedge \sigma_1 \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][\bar{S} \to \bullet\varsigma]\}$ ≀since $\epsilon = \sigma_1 \ldots \sigma_i \Longleftrightarrow i = 0$≀

$= \{\langle 0,\ \dashv[\bar{S} \to \bullet\varsigma]\rangle \mid \langle 0,\ \vdash\rangle \in \alpha^{LL(1)}(\bar{S})(\sigma)(X) \wedge \bar{S} \to \varsigma \in \mathcal{R} \wedge \sigma_1 \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][\bar{S} \to \bullet\varsigma]\}$ ≀def. $\alpha^{LL(1)}$≀

— $\alpha^{LL(1)}(\bar{S})(\sigma)(B)$

$= \alpha^{LL(1)}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta\bullet a\eta'] \xrightarrow{a} \varpi[A \to \eta a\bullet\eta'] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta\bullet a\eta'] \in X \wedge A \to \sigma a\sigma' \in \mathcal{R}\})$ ≀def. case (B)≀

$= \alpha^{LL(1)}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta\bullet a\eta'] \xrightarrow{a} \varpi[A \to \eta a\bullet\eta'] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta\bullet a\eta'] \in X\})$

≀because X is an iterate of $\vec{\mathsf{F}}^{\partial}[\![\mathcal{G}]\!]$ so, by Lemma 7, $[A \to \eta\bullet a\eta']$ can be on the stack only if $A \to \sigma a\sigma'$ is a grammar rule in $\mathcal{R}$≀

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta\bullet a\eta'] \xrightarrow{a} \varpi'[A \to \eta a\bullet\eta'] \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta\bullet a\eta'] \in X.\bar{S}\} : i \in [0, |\sigma|] \wedge \alpha^{\tau}(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m \wedge \forall \varpi' \in \mathcal{S}, A \to \eta\eta' \in \mathcal{R} : (\varpi = \varpi'[A \to \eta\bullet\eta'] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}[\![\mathcal{G}]\!][A \to \eta\bullet\eta'])\}$ ≀def. $\alpha^{LL(1)}(\bar{S})(\sigma)$ and selection $\bullet.\bar{S}$≀

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \in X.\overline{S}, \varpi_{m-1} = \varpi'[A \to \eta.a\eta'], \ell_{m-1} = a, \varpi_m = \varpi'[A \to \eta a.\eta'] : m \geqslant 1 \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'' \xrightarrow{\ell_{m-1}} \varpi_m) = \sigma_1\ldots\sigma_i \wedge \varpi = \varpi_m \wedge \forall\varpi' \in \mathscr{S}, A \to \eta\eta' \in \mathscr{R} : (\varpi = \varpi'[A \to \eta.\eta'] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta.\eta'])\}$ ⟅def. $\in$ with $\theta = \theta'' \xrightarrow{\ell_{m-1}} \varpi_m$⟆

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \in X.\overline{S}, \varpi_{m-1} = \varpi'[A \to \eta.a\eta'], \ell_{m-1} = a, \varpi_m = \varpi'[A \to \eta a.\eta'] : m \geqslant 1 \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'' \xrightarrow{\ell_{m-1}} \varpi_m) = \sigma_1\ldots\sigma_i \wedge \varpi = \varpi_m \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$
⟅since $\varpi = \varpi_m = \varpi'[A \to \eta a.\eta']$ so $\sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']$⟆

$= \{\langle i,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_m \in X.\overline{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'')a = \sigma_1\ldots\sigma_i \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅def. α^τ and setting the dummy variable m to $m-1 \geqslant 0$⟆

$= \{\langle i,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_m \in X.\overline{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [1, |\sigma|] \wedge \alpha^\tau(\theta'')a = \sigma_1\ldots\sigma_i \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅since $\alpha^\tau(\theta'')a = \sigma_1\ldots\sigma_i$ implies $1 \leqslant i \leqslant |\sigma|$⟆

$= \{\langle i+1,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_m \in X.\overline{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [0, |\sigma|-1] \wedge \alpha^\tau(\theta'')a = \sigma_1\ldots\sigma_{i+1} \wedge \sigma_{i+2} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅setting the dummy variable i to $i+1$⟆

$= \{\langle i+1,\ \varpi'[A \to \eta a.\eta']\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_m \in X.\overline{S}, \varpi_m = \varpi'[A \to \eta.a\eta'] : i \in [0, |\sigma|-1] \wedge \alpha^\tau(\theta'') = \sigma_1\ldots\sigma_i \wedge \sigma_{i+1} = a \wedge \sigma_{i+2} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅def. equality of sequences⟆

$= \{\langle i+1,\ \varpi[A \to \eta a.\eta']\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\overline{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1\ldots\sigma_i \wedge \varpi[A \to \eta.a\eta'] = \varpi_m \wedge a = \sigma_{i+1} \wedge \sigma_{i+2} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅since $\sigma_{i+1} = a$ implies $i+1 \leqslant |\sigma|$⟆

$= \{\langle i+1,\ \varpi[A \to \eta a.\eta']\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\overline{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1\ldots\sigma_i \wedge \varpi[A \to \eta.a\eta'] = \varpi_m \wedge a = \sigma_{i+1} \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta.a\eta'] \wedge \sigma_{i+2} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅since $\overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta.a\eta'] = \{a\} = \{\sigma_{i+1}\}$⟆

$= \{\langle i+1,\ \varpi[A \to \eta a.\eta']\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\overline{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1\ldots\sigma_i \wedge \varpi[A \to \eta.a\eta'] = \varpi_m \wedge a = \sigma_{i+1} \wedge \forall\varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi_m = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A' \to \eta''.\eta''']) \wedge \sigma_{i+2} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅with $A' = A$, $\eta'' = \eta$ and $\eta''' = \eta'' a$ since $\varpi_m = \varpi[A \to \eta.a\eta']$⟆

$= \{\langle i+1,\ \varpi[A \to \eta a.\eta']\rangle \mid \langle i,\ \varpi[A \to \eta.a\eta']\rangle \in \alpha^{LL(1)}(\overline{S})(\sigma)(X) \wedge a = \sigma_{i+1} \wedge \sigma_{i+2} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A \to \eta a.\eta']\}$ ⟅def. $\in$ and $\alpha^{LL(1)}(\overline{S})(\sigma)$⟆

— $\alpha^{LL(1)}(\overline{S})(\sigma)(C)$

$= \alpha^{LL(1)}(\overline{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \xrightarrow{(\!|B} \varpi[A \to \eta B.\eta'][B \to .\varsigma] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \in X \wedge A \to \sigma B\sigma' \in \mathscr{R} \wedge B \to \varsigma \in \mathscr{R}\})$ ⟅def. case (C)⟆

$= \alpha^{LL(1)}(\overline{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \xrightarrow{(\!|B} \varpi[A \to \eta B.\eta'][B \to .\varsigma] \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.B\eta'] \in X \wedge B \to \varsigma \in \mathscr{R}\})$
⟅because X is an iterate of $\overrightarrow{\mathsf{F}}^{\partial}[\![\mathcal{G}]\!]$ so by Lemma 7, $[A \to \eta.B\eta']$ can be on the stack only if $A \to \sigma B\sigma'$ is a grammar rule in $\mathscr{R}$⟆

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.B\eta'] \xrightarrow{(\!|B} \varpi'[A \to \eta B.\eta'][B \to .\varsigma] \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.B\eta'] \in X.\overline{S} \wedge B \to \varsigma \in \mathscr{R}\} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1\ldots\sigma_i \wedge \varpi = \varpi_m \wedge \forall\varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A' \to \eta''.\eta'''])\}$ ⟅def. $\alpha^{LL(1)}(\overline{S})(\sigma)$ and selection $\bullet.\overline{S}$⟆

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \in X.\overline{S}, \varpi_{m-1} = \varpi'[A \to \eta.B\eta'], \ell_{m-1} = (\!|B, \varpi_m = \varpi'[A \to \eta B.\eta'][B \to .\varsigma] : m \geqslant 1 \wedge B \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'' \xrightarrow{\ell_{m-1}} \varpi_m) = \sigma_1\ldots\sigma_i \wedge \varpi = \varpi_m \wedge \forall\varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][A' \to \eta''.\eta'''])\}$ ⟅def. $\in$ and $\theta = \theta'' \xrightarrow{\ell_{m-1}} \varpi_m$⟆

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \in X.\overline{S}, \varpi_{m-1} = \varpi'[A \to \eta.B\eta'], \ell_{m-1} = (\!|B, \varpi_m = \varpi'[A \to \eta B.\eta'][B \to .\varsigma] : m \geqslant 1 \wedge B \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'' \xrightarrow{\ell_{m-1}} \varpi_m) = \sigma_1\ldots\sigma_i \wedge \varpi = \varpi_m \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][B \to .\varsigma]\}$
⟅since $\varpi = \varpi_{m-1} = \varpi'[A \to \eta B.\eta'][B \to .\varsigma]$ so $\varpi'' = \varpi'[A \to \eta B.\eta']$, $A' = B$, $\eta'' = \epsilon$ and $\eta''' = \varsigma$⟆

$= \{\langle i,\ \varpi'[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \in X.\overline{S}, \varpi_{m-1} = \varpi'[A \to \eta.B\eta'] : m \geqslant 1 \wedge B \to \varsigma \in \mathscr{R} \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'') = \sigma_1\ldots\sigma_i \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][B \to .\varsigma]\}$ ⟅def. α^τ⟆

$= \{\langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1\ldots\varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\overline{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1\ldots\sigma_i \wedge \varpi[A \to \eta.B\eta'] = \varpi_m \wedge B \to \varsigma \in \mathscr{R} \wedge \sigma_{i+1} \in \overrightarrow{\mathsf{S}}^1[\![\mathcal{G}]\!][B \to .\varsigma]\}$ ⟅setting the dummy variable m to $m-1 \geqslant 0$ and $\theta = \theta''$⟆

$= \{\langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.B\eta'] = \varpi_m \wedge B \to \varsigma \in \mathscr{R} \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A \to \eta.B\eta'] \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[B \to .\varsigma]\}$

$\wr$since $\vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A \to \eta.B\eta'] = \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[B \to .\varsigma]$ by def. (60) of $\vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket\wr$

$= \{\langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.B\eta'] = \varpi_m \wedge B \to \varsigma \in \mathscr{R} \wedge \forall \varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi_m = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta''']) \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[B \to .\varsigma]\}$ $\wr$since $\varpi_m = \varpi[A \to \eta.B\eta']$ so that $A' = A$, $\eta'' = \eta$ and $\eta''' = B\eta'\wr$

$= \{\langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle \mid \langle i,\ \varpi[A \to \eta.B\eta']\rangle \in \alpha^{LL(1)}(\bar{S})(\sigma)(X) \wedge B \to \varsigma \in \mathscr{R} \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[B \to .\varsigma]\}$ $\wr$def. $\in$ and $\alpha^{LL(1)}(\bar{S})(\sigma)\wr$

— $\alpha^{LL(1)}(\bar{S})(\sigma)(D)$

$= \alpha^{LL(1)}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.] \xrightarrow{A\rrparenthesis} \varpi \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.] \in X \wedge A \to \eta \in \mathscr{R}\})$

$\wr$def. case (D)$\wr$

$= \alpha^{LL(1)}(\bar{S})(\sigma)(\{\theta \xrightarrow{\ell} \varpi[A \to \eta.] \xrightarrow{A\rrparenthesis} \varpi \mid \theta \xrightarrow{\ell} \varpi[A \to \eta.] \in X\})$

$\wr$because X is an iterate of $\vec{\mathsf{F}}^{\partial}\llbracket\mathscr{G}\rrbracket$ so, by Lemma 7, $[A \to \eta.]$ can be on the stack only if $A \to \eta$ is a grammar rule in $\mathscr{R}\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \xrightarrow{A\rrparenthesis} \varpi' \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \in X.\bar{S}\} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi = \varpi_m \wedge \forall \varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta'''])\}$ $\wr$def. $\alpha^{LL(1)}(\bar{S})(\sigma)$ and selection $\bullet.\bar{S}\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta'' = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \in \{\theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \mid \theta' \xrightarrow{\ell} \varpi'[A \to \eta.] \in X.\bar{S}\} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta'') = \sigma_1 \ldots \sigma_i \wedge \varpi_{m-1} = \varpi[A \to \eta.] \wedge \forall \varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta'''])\}$ $\wr$setting $\theta = \theta'' \xrightarrow{\ell_{m-1}} \varpi_m$ with $\ell_{m-1} = A\rrparenthesis$, $\varpi_m = \varpi$ and $\varpi_{m-1} = \varpi[A \to \eta.]$ since $\alpha^\tau(\theta) = \alpha^\tau(\theta'')\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \in X.\bar{S} : m \geqslant 1 \wedge i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.] = \varpi_{m-1} \wedge \forall \varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta'''])\}$ $\wr$def. $\in\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.] = \varpi_m \wedge \forall \varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta'''])\}$ $\wr$setting the dummy variable m to $m - 1 \geqslant 0\wr$

$= \{\langle i,\ \varpi\rangle \mid \exists\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S} : i \in [0, |\sigma|] \wedge \alpha^\tau(\theta) = \sigma_1 \ldots \sigma_i \wedge \varpi[A \to \eta.] = \varpi_m \wedge \forall \varpi'' \in \mathscr{S}, A' \to \eta''\eta''' \in \mathscr{R} : (\varpi[A \to \eta.] = \varpi''[A' \to \eta''.\eta'''] \wedge i \leqslant |\sigma|) \Longrightarrow (\sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta'''])\}$

$\wr$since $\theta = \varpi_0 \xrightarrow{\ell_0} \varpi_1 \ldots \varpi_{m-1} \xrightarrow{\ell_{m-1}} \varpi_m \in X.\bar{S}$ so that by Lemma 7, $\varpi_m = \varpi[A \to \eta.] = \dashv[A_1 \to \eta_1 A_2.\eta'_1][A_2 \to \eta_2 A_3.\eta'_2] \ldots [A_{n-1} \to \eta_{n-1}A.\eta'_{n-1}][A \to \eta.]$ and therefore $\varpi = \dashv[A_1 \to \eta_1 A_2.\eta'_1][A_2 \to \eta_2 A_3.\eta'_2] \ldots [A_{n-1} \to \eta_{n-1}A.\eta'_{n-1}]$ with $A_{n-1} \to \eta_{n-1}A\eta'_{n-1}$, $[A \to \eta.] \in \mathscr{R}$ that is necessarily $\varpi'' = \dashv[A_1 \to \eta_1 A_2.\eta'_1][A_2 \to \eta_2 A_3.\eta'_2] \ldots$ and $[A' \to \eta''.\eta'''] = [A_{n-1} \to \eta_{n-1}A.\eta'_{n-1}]$ so $\vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A' \to \eta''.\eta'''] = \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A_{n-1} \to \eta_{n-1}A.\eta'_{n-1}] = \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A \to \eta.]$ by def. (60) of $\vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket\wr$

$= \{\langle i,\ \varpi\rangle \mid \langle i,\ \varpi[A \to \eta.]\rangle \in \alpha^{LL(1)}(\bar{S})(\sigma)(X)\}$ $\wr$def. $\in$ and $\alpha^{LL(1)}(\bar{S})(\sigma)\wr$. $\square$

Again, observe that, by Example 107, $\mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL(1)}\llbracket\mathscr{G}\rrbracket(\sigma)$ is exactly the set of reachable states of the transition system $\langle [0, |\sigma|] \times \mathscr{S}, \xrightarrow{\text{LL(1)}}\rangle$ where

$$
\begin{array}{rll}
\langle 0,\ \vdash\rangle \xrightarrow{\text{LL(1)}} \langle 0,\ \dashv[\bar{S} \to .\eta]\rangle & & \bar{S} \to \eta \in \mathscr{R} \wedge \sigma_1 \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[\bar{S} \to .\eta] \\
\langle i,\ \varpi[A \to \eta.\sigma_{i+1}\eta']\rangle \xrightarrow{\text{LL(1)}} \langle i+1,\ \varpi[A \to \eta\sigma_{i+1}.\eta']\rangle & & \sigma_{i+2} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[A \to \eta a.\eta'] \\
\langle i,\ \varpi[A \to \eta.B\eta']\rangle \xrightarrow{\text{LL(1)}} \langle i,\ \varpi[A \to \eta B.\eta'][B \to .\varsigma]\rangle & & B \to \varsigma \in \mathscr{R} \wedge \sigma_{i+1} \in \vec{\mathsf{S}}^{1}\llbracket\mathscr{G}\rrbracket[B \to .\varsigma]\} \\
\langle i,\ \varpi[A \to \eta.]\rangle \xrightarrow{\text{LL(1)}} \langle i,\ \varpi\rangle & &
\end{array}
$$

with initial state $\langle 0,\ \vdash\rangle$. This is essentially the algorithm suggested at the end of [4, Section 4.1.4] to speed up top-down nondeterministic parsing.

Indeed the lookahead may been done freely between the two extremes of everywhere in Theorem 94 and nowhere Theorem 88, as follows

Corollary 96. *If* $\mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma) \subseteq \mathsf{F}[\![\mathcal{G}]\!](\sigma) \subseteq \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ *then*

$$\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S}) \quad\Longleftrightarrow\quad \langle|\sigma|,\ \dashv\rangle \in \mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma).$$

The iterative computation of $\mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma)$ *terminates for all* σ *if and only if the grammar* $\mathcal{G}$ *has no left recursion.*

Proof. — We have $\mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma) \subseteq \mathsf{F}[\![\mathcal{G}]\!](\sigma) \subseteq \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ so, by Corollary 102, $\mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma) \subseteq \mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma) \subseteq \mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$.

It follows that $\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S})$ implies $\langle|\sigma|,\ \dashv\rangle \in \alpha^{LL(1)}(\bar{S})(\sigma)(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!])$ by Theorem 94 and therefore $\langle|\sigma|,\ \dashv\rangle \in \mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma)$ by Theorem 95 whence $\langle|\sigma|,\ \dashv\rangle \in \mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma)$.

Reciprocally, $\langle|\sigma|, \dashv\rangle \in \mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma)$ implies $\langle|\sigma|, \dashv\rangle \in \mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ whence $\langle|\sigma|, \dashv\rangle \in \alpha^{LL}(\bar{S})(\sigma)(\mathsf{S}^{\vec{\partial}}[\![\mathcal{G}]\!])$ by Theorem 89 so $\sigma \in \mathsf{S}^{\ell}[\![\mathcal{G}]\!](\bar{S})$ by Theorem 88.

— If the grammar has no left recursion then by Theorem 91, $\mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$ has only finite traces whence so has $\mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma) \subseteq \mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$.

Reciprocally, if the grammar is left-recursive, then by Theorem 91, there is an infinite trace in $\mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma) \subseteq \mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL}[\![\mathcal{G}]\!](\sigma)$. To show that it is also in $\mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma)$, it is sufficient to shown that it is in $\mathbf{lfp}^{\subseteq}\ \mathsf{F}^{LL(1)}[\![\mathcal{G}]\!](\sigma) \subseteq \mathbf{lfp}^{\subseteq}\ \mathsf{F}[\![\mathcal{G}]\!](\sigma)$ which follows from the fact that the lookahead conditions prevent none of these transitions by Corollary 93. □

23.6. Correspondence with the classical nonrecursive predictive parsing algorithm

Our presentation of LL(1) parsing differs from the classical introduction in [32] or [8], mainly because, for practical efficiency and simplicity reasons, only the table-driven deterministic case is classically considered.

24. Conclusion

Many meanings assigned to grammars (such as syntax tree, protolanguage or terminal language generation) and grammar manipulation algorithms (such as grammar flow analyses or parsers) have quite similar structures. We have shown that this is because they are all abstract interpretations of a grammar small-step operational semantics to derive sentences together with their structure.

The verification of compilers is an old and challenging problem [37] which has recently made significant progress [38, 39]. Indeed [37] originated the use of abstract syntax in order to get rid of the concrete parsing problem. Having formalized parsing by abstract interpretation, one can hope that the parser correctness can be integrated in the full compiler correctness proof, together with the validity of the concrete to abstract syntax translation. Because abstraction can be constructed by calculational design [40], as shown in our formal proofs, proof assistant or theorem provers can be used to automatically check or perform these calculations. This has been done for simple abstract interpreters in restricted cases excluding the use of Galois connections [41], whence some progress in automatic verification/proof checking is still needed before this paper can be entirely checked mechanically, which is the ultimate "proof by construction" goal in abstract-interpretation-based designs.

The results obtained in this paper directly extend to the semantics and static analysis of resolution-based languages [42]. Future work should include the extension of the approach to context-sensitive grammars such as *contextual grammars* [43, 44] or to mildly context-sensitive grammars attempting to express the formal power needed to define the syntax of natural languages by tree rewriting such as (multicomponent) tree adjoining grammars or, more generally, *range concatenation grammars* [45].

Acknowledgement

We thank Tom Reps for drawing our attention to [1,2].

Appendix A

A.1. Posets, Booleans, maps, iteration and fixpoints

A *poset* $\langle P, \preceq\rangle$ is a set P equipped with a partial order $\preceq$ [46]. If $X \subseteq P$ then $\curlyvee X$ denotes the least upper bound (lub) of X and $\curlywedge X$ denotes its greatest lower bound (glb), if any. A *complete lattice* has all lubs whence all glbs, an infimum 0 and a supremum 1. A *complete Boolean* lattice is a complete lattice with unique *complement* $\neg$ (i.e. $\forall x \in P : (x \curlyvee \neg x = 1) \wedge (x \curlywedge \neg x = 0)$).

We let $\mathbb{B} \triangleq \{\mathbb{ff}, \mathbb{tt}\}$ where $\mathbb{ff}$ is false $\mathbb{tt}$ is true be the Booleans ordered by *implication* $\mathbb{ff} \Longrightarrow \mathbb{ff} \Longrightarrow \mathbb{tt} \Longrightarrow \mathbb{tt}$. It is a complete Boolean lattice $\langle \mathbb{B},\ \Longrightarrow,\ \mathbb{ff},\ \mathbb{tt},\ \vee,\ \wedge,\ \neg\rangle$. The *conditional* $(\!|b\ ?\ x\ ⨾\ y|\!)$ is x if b holds and y otherwise that is

$(\!\!\lbrack \mathrm{tt} \mathrel{?} x \mathrel{\S} y \rbrack\!\!) = x$ and $(\!\!\lbrack \mathrm{ff} \mathrel{?} x \mathrel{\S} y \rbrack\!\!) = y$. We sometimes write $(\!\!\lbrack b \mathrel{?} \mathrm{tt} \mathrel{\S} \mathrm{ff} \rbrack\!\!)$ for b, a redundancy emphasizing the computer Boolean encoding of b.

If $\langle Q, \sqsubseteq, \sqcup\rangle$ is a poset, we say that the map $f \in P \mapsto Q$ is *monotone* if and only if $\forall x, y \in P : (x \preceq y) \Longrightarrow (f(x) \sqsubseteq f(y))$. f is *lub-preserving* whenever the existence of $\curlyvee_\beta x^\beta$ in P implies the existence of $\bigsqcup_\beta f(x^\beta)$ in Q such that $f(\curlyvee_\beta x^\beta) = \bigsqcup_\beta f(x^\beta)$. f is *upper continuous* (*continuous* for short) if and only if it preserves existing lubs of increasing denumerable chains x_n, $n \in \mathbb{N}$, that is if $\forall n \in \mathbb{N} : x_n \preceq x_{n+1}$ and the lub $\curlyvee_{n\in\mathbb{N}} x_n$ does exist then $\bigsqcup_{n\in\mathbb{N}} f(x_n)$ exists such that $f(\curlyvee_{n\in\mathbb{N}} x_n) = \bigsqcup_{n\in\mathbb{N}} f(x_n)$.

The transfinite *iterates of* $F \in P \mapsto P$ *from* $a \in P$ are partially defined as $F^0 \triangleq a$, $F^{\delta+1} \triangleq F(F^\delta)$ for successor ordinals and $F^\lambda \triangleq \curlyvee_{\beta<\lambda} F^\beta$ for limit ordinals λ [28]. This is well defined only when the lubs $\curlyvee$ do exist in $\langle P, \preceq\rangle$.

If $\langle P, \preceq\rangle$ is a partial order and $F \in P \mapsto P$ then $\mathbf{lfp}^{\preceq} F$ denotes the *least fixpoint* of F on P, if any, that is $F(\mathbf{lfp}^{\preceq} F) = \mathbf{lfp}^{\preceq} F$ and $\forall x \in P : F(x) = x \Longrightarrow \mathbf{lfp}^{\preceq} F \preceq x$. If P has an infimum $\perp$, F is continuous (in particular F preserves existing lubs) and the iterates of F from $\perp$ have a lub F^ω then $F^\omega = \mathbf{lfp}^{\preceq} F$ [8, Section 8.2.5]. Hereafter we use the notation $\mathbf{lfp}^{\preceq} F$ only when it exists (most often because $\langle P, \preceq\rangle$ is a complete lattice and F preserves lubs or is continuous [28]).

A.2. Abstraction and fixpoint abstraction

In this paper, all abstract interpretations [27] use *Galois connections* $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle Q, \sqsubseteq\rangle$ that is, by definition, $\langle P, \preceq\rangle$ and $\langle Q, \sqsubseteq\rangle$ are posets, $\alpha \in P \mapsto Q$ and $\gamma \in Q \mapsto P$ satisfy $\forall x \in P : \forall y \in Q : \alpha(x) \sqsubseteq y \Longleftrightarrow x \preceq \gamma(y)$. It follows that α preserves lubs existing in P and, by duality, γ preserves greatest glbs existing in Q. Given a lub-preserving α (resp. glb-preserving γ), there exists a unique γ (resp. α) such that $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle Q, \sqsubseteq\rangle$. α is onto if and only if γ is one-to-one, written $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\leftarrows\!\!\!\twoheadrightarrow}} \langle Q, \sqsubseteq\rangle$. Dually, γ is onto if and only if α is one-to-one, written $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\twoheadleftarrow\!\!\!\rightarrow}} \langle Q, \sqsubseteq\rangle$. A Galois isomorphism is written $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\twoheadleftarrow\!\!\!\twoheadrightarrow}} \langle Q, \sqsubseteq\rangle$.

Example 97 (*Function Abstraction at a Point*)**.** If $\langle L, \sqsubseteq, \top\rangle$ is a poset $\langle L, \sqsubseteq\rangle$ with supremum $\top$ and $x \in L$ then we define the abstraction of functions in $L \mapsto L$ at point x by $\alpha^x \triangleq \boldsymbol{\lambda} f \bullet f(x)$ and $\gamma^x \triangleq \boldsymbol{\lambda} v \bullet \boldsymbol{\lambda} s \bullet (\!\!\lbrack s = x \mathrel{?} v \mathrel{\S} \top \rbrack\!\!)$. We have $\langle L \mapsto L, \sqsubseteq\rangle \underset{\alpha^x}{\overset{\gamma^x}{\leftrightarrows}} \langle L, \sqsubseteq\rangle$.

Proof. For all $f \in L \mapsto L$ and $v \in L$, we have $\alpha^x(f) \sqsubseteq v$

$\Longleftrightarrow \quad \forall s \in L : f(s) \sqsubseteq (\!\!\lbrack s = x \mathrel{?} v \mathrel{\S} \top \rbrack\!\!)$ $\qquad$ ⟨def. α^x and $\top$ is the supremum of L⟩

$\Longleftrightarrow \quad f \sqsubseteq \gamma^x(v)$ $\qquad$ ⟨def. pointwise ordering and γ^x⟩. □ □

Let $\circ$ be the composition of relations or functions. The composition of Galois connections $\langle P, \preceq\rangle \underset{\alpha_1}{\overset{\gamma_1}{\leftrightarrows}} \langle Q, \sqsubseteq\rangle$ and $\langle Q, \sqsubseteq\rangle \underset{\alpha_2}{\overset{\gamma_2}{\leftrightarrows}} \langle R, \leqslant\rangle$ is a Galois connection $\langle P, \preceq\rangle \underset{\alpha_2\circ\alpha_1}{\overset{\gamma_1\circ\gamma_2}{\leftrightarrows}} \langle R, \leqslant\rangle$.

We use a weaker variant of the fixpoint abstraction theorem [27, Th. 7.1.0.4(3)] as follows

Theorem 98. *If $\langle P, \preceq, 0, \vee\rangle$ is a poset with infimum 0, $F \in P \mapsto P$ is monotone, the iterates of F from 0 are well defined with iteration order ϵ, $\langle Q, \sqsubseteq, \perp, \sqcup\rangle$ is a poset with infimum $\perp$, $F^\sharp \in Q \mapsto Q$ is monotone, the iterates of $F^\sharp$ from $\perp$ are well defined with iteration order $\epsilon^\sharp$, then $\mathbf{lfp}^{\preceq} F$ and $\mathbf{lfp}^{\sqsubseteq} F^\sharp$ do exist. Moreover, if for all ordinals $\delta \in \mathbb{O}$, the maps $\alpha_\delta \in P \mapsto Q$ satisfy the* correspondence property

$$\forall \delta : \alpha_\delta(F^\delta) \quad \sqsubseteq\!\!\!\!\sqsupseteq \quad F^{\sharp\delta},$$

where $\sqsubseteq\!\!\!\!\sqsupseteq$ denotes either $\sqsubset$, $\sqsubseteq$, $=$, $\sqsupseteq$ or $\sqsupset$, then

$$\alpha_{\max(\epsilon,\epsilon^\sharp)}(\mathbf{lfp}^{\preceq} F) \quad \sqsubseteq\!\!\!\!\sqsupseteq \quad \mathbf{lfp}^{\sqsubseteq} F^\sharp.$$

Proof. By monotony and well-definedness, the iterates of F form an increasing chain, ultimately stationary at rank ϵ, with lub $F^\epsilon = \mathbf{lfp}^{\preceq} F$ [28]. Similarly, the iterates of $F^\sharp$ form an increasing chain, ultimately stationary at rank $\epsilon^\sharp$, with lub $F^{\sharp\epsilon^\sharp} = \mathbf{lfp}^{\sqsubseteq} F^\sharp$ [28].

By stationarity, we have $\alpha_{\max(\epsilon,\epsilon^\sharp)}(\mathbf{lfp}^{\preceq} F) = \alpha_{\max(\epsilon,\epsilon^\sharp)}(F^\epsilon) = \alpha_{\max(\epsilon,\epsilon^\sharp)}(F^{\max(\epsilon,\epsilon^\sharp)}) \sqsubseteq\!\!\!\!\sqsupseteq F^{\sharp\max(\epsilon,\epsilon^\sharp)} = F^{\sharp\epsilon^\sharp} = \mathbf{lfp}^{\sqsubseteq} F^\sharp$.

Corollary 99. *If $\langle P, \preceq, 0, \vee\rangle$ is a poset with infimum 0, $F \in P \mapsto P$ is monotone, the iterates of F from 0 are well defined with iteration order ϵ, $\langle Q, \sqsubseteq, \perp, \sqcup\rangle$ is a poset with infimum $\perp$, $F^\sharp \in Q \mapsto Q$ is monotone, the iterates of $F^\sharp$ from $\perp$ are well defined with iteration order $\epsilon^\sharp$ then $\mathbf{lfp}^{\preceq} F$ and $\mathbf{lfp}^{\sqsubseteq} F^\sharp$ do exist. Moreover, if, for all ordinals $\delta \in \mathbb{O}$, the maps $\alpha_\delta \in P \mapsto Q$ satisfy the* commutation property

$$\forall \delta \in \mathbb{O} : \alpha_{\delta+1} \circ F(F^\delta) \quad \sqsubseteq\!\!\!\!\sqsupseteq \quad F^\sharp \circ \alpha_\delta(F^\delta),$$

where $\sqsubseteq\!\!\!\!\!\!\;\sqsupseteq$ …

where $\overline{\sqsubseteq}$ *denotes either* $\sqsubseteq$, $=$ *or* $\sqsupseteq$, $\alpha_0(0) \overline{\sqsubseteq} \bot$ *and for all limit ordinals* λ, $\alpha_\lambda(\bigvee_{\beta<\lambda} F^\beta) \overline{\sqsubseteq} \bigsqcup_{\beta<\lambda} \alpha_\beta(F^\beta)$ *then* $\forall \delta : \alpha_\delta(F^\delta) \overline{\sqsubseteq} F^{\sharp^\delta}$ *and*

$$\alpha_{\max(\epsilon,\epsilon^\sharp)}(\mathbf{lfp}^{\preceq} F) \quad \overline{\sqsubseteq} \quad \mathbf{lfp}^{\sqsubseteq} F^\sharp.$$

Proof. By monotony and well-definedness, the iterates of F form an increasing chain, ultimately stationary at rank ϵ, with lub $F^\epsilon = \mathbf{lfp}^{\preceq} F$ [28]. Similarly, the iterates of $F^\sharp$ form an increasing chain, ultimately stationary at rank $\epsilon^\sharp$, with lub $F^{\sharp^{\epsilon^\sharp}} = \mathbf{lfp}^{\sqsubseteq} F^\sharp$ [28].

We have $\alpha_0(F^0) = \alpha_0(0) \overline{\sqsubseteq} \bot = F^{\sharp^0}$. Assuming $\alpha_\delta(F^\delta) \overline{\sqsubseteq} F^{\delta^0}$ by induction hypothesis, we have $\alpha_{\delta+1}(F^{\delta+1})$

$$\begin{aligned}
&= \alpha_{\delta+1}(F(F^\delta)) \quad \overline{\sqsubseteq} \quad F^\sharp(\alpha_\delta(F^\delta)) && \text{(def. iterates and commutation hyp.)}\\
&\overline{\sqsubseteq} \ F^\sharp(F^{\sharp^\delta}) && \text{(ind. hyp. \& monotony of } F^\sharp \text{ (when } \overline{\sqsubseteq} \text{ is, } \sqsubseteq \text{ or } \sqsupseteq\text{) or equality)}\\
&= F^{\sharp^{\delta+1}} && \text{(def. iterates.)}
\end{aligned}$$

For limit ordinals, $\alpha_\lambda(F^\lambda)$

$$\begin{aligned}
&= \alpha_\lambda\left(\bigvee_{\beta<\lambda} F^\beta\right) \overline{\sqsubseteq} \bigsqcup_{\beta<\lambda} \alpha_\beta(F^\beta) && \text{(def. iterates and lub approximation hypothesis)}\\
&\overline{\sqsubseteq} \ \bigsqcup_{\beta<\lambda} F^{\sharp^\beta} && \text{(ind. hyp. \& monotony of the lub)}\\
&= F^{\sharp^\lambda} && \text{(def. well-defined iterates (so the lub exists))}
\end{aligned}$$

We proved $\forall \delta : \alpha_\delta(F^\delta) \overline{\sqsubseteq} F^{\sharp^\delta}$ and conclude by Theorem 98. □

Note that we may have $\epsilon^\sharp > \epsilon$ as in $P = \{0\}$, $F(0) = 0$ so $\epsilon = 0$, $Q = \{\bot, \top\}$ with $\bot \preceq \bot \prec \top \preceq \top$, $F^\sharp(\bot) = F^\sharp(\top) = \top$ so $\epsilon^\sharp = 1$, $\alpha_0(0) = \bot$ and $\alpha_1(0) = \top$.

Corollary 100. *Corollary 99 holds with the stronger* commutation property

$$\forall x \in P : x \preceq \mathbf{lfp}^{\preceq} F \Longrightarrow \alpha_{\delta+1} \circ F(x) \overline{\sqsubseteq} F^\sharp \circ \alpha_\delta(x). \quad \square$$

Corollary 101. *If* $\langle P, \preceq, 0, \vee\rangle$ *is a poset with infimum* 0, $F \in P \mapsto P$ *is monotone, the iterates of* F *are well defined with iteration order* ϵ, $\langle Q, \sqsubseteq, \sqcup\rangle$ *is a poset,* $F^\sharp \in Q \mapsto Q$ *is monotone, the Galois connection* $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle Q, \sqsubseteq\rangle$ *satisfy the* commutation property

$$\forall \delta \in \mathbb{O} : \alpha \circ F(F^\delta) \overline{\sqsubseteq} F^\sharp \circ \alpha(F^\delta),$$

where $\overline{\sqsubseteq}$ *denotes either* $\sqsubseteq$, $=$ *or* $\sqsupseteq$, *then* $\forall \delta \in \mathbb{O} : \alpha(F^\delta) \overline{\sqsubseteq} F^{\sharp^\delta}$ *and* $\mathbf{lfp}^{\sqsubseteq} F^\sharp$ *does exist such that*

$$\alpha(\mathbf{lfp}^{\preceq} F) \quad \overline{\sqsubseteq} \quad \mathbf{lfp}^{\sqsubseteq} F^\sharp.$$

Proof. We apply Corollary 99 with $\forall \delta \in \mathbb{O} : \alpha_\delta = \alpha$. $\langle P, \preceq\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle Q, \sqsubseteq\rangle$ implies that $\alpha(0)$ is the infimum $\bot$ of Q and α preserves existing lubs, so the iterates of $F^\sharp$ do exist and for all limit ordinals λ, $\alpha_\lambda(\bigvee_{\beta<\lambda} F^\beta) \overline{\sqsubseteq} \bigsqcup_{\beta<\lambda} \alpha_\beta(F^\beta)$ by reflexivity of $\overline{\sqsubseteq}$. □

Corollary 102. *If* F *and* G *are monotone transformers on a cpo* $\langle P, \preceq, 0, \vee\rangle$ *and* $F \preceq G$ *pointwise, then* $\mathbf{lfp}^{\preceq} F \preceq \mathbf{lfp}^{\preceq} G$. □

Proof. By Corollary 101 with $\alpha = \mathbb{1}_P$. □

Example 103 (*Common Least Fixpoint*)**.** If F is monotone on a cpo then $\mathbf{lfp}^{\preceq} F = \mathbf{lfp}^{\preceq} \boldsymbol{\lambda} X \bullet X \sqcup F(X)$.

Proof. $\mathbf{lfp}^{\preceq} F$ is a fixpoint of $\boldsymbol{\lambda} X \bullet X \sqcup F(X)$ so $\mathbf{lfp}^{\preceq} \boldsymbol{\lambda} X \bullet X \sqcup F(X) \preceq \mathbf{lfp}^{\preceq} F$. $F \preceq \boldsymbol{\lambda} X \bullet X \sqcup F(X)$ pointwise so by Corollary 102 $\mathbf{lfp}^{\preceq} F \preceq \mathbf{lfp}^{\preceq} \boldsymbol{\lambda} X \bullet X \sqcup F(X)$. We conclude by antisymmetry. □

In the particular case when $\overline{\sqsubseteq}$ is $=$, we can weaken the hypotheses in Corollary 99 as follows

Corollary 104. *If $\langle P, \preceq, 0, \vee\rangle$ is a poset with infimum 0, $F \in P \mapsto P$ is monotone, the iterates of F are well defined with iteration order ϵ, $\langle Q, \sqsubseteq, \sqcup\rangle$ is a poset, $F^\sharp \in Q \mapsto Q$, the Galois connection $\langle P, \preceq\rangle \xleftarrow[\alpha]{\gamma} \langle Q, \sqsubseteq\rangle$ satisfy the* commutation property

$$\forall \delta \in \mathbb{O} : \alpha \circ F(F^\delta) = F^\sharp \circ \alpha(F^\delta),$$

then $\forall \delta \in \mathbb{O} : \alpha(F^\delta) = F^{\sharp^\delta}$ and

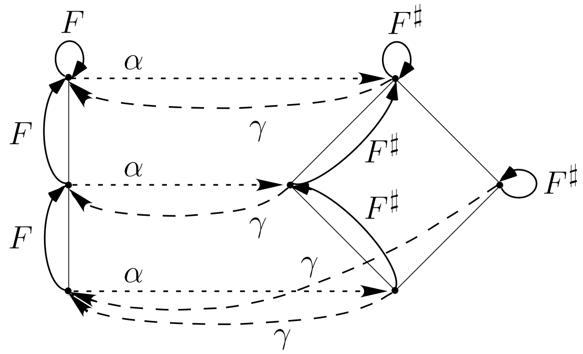
$$\alpha(\mathbf{lfp}^{\preceq} F) \quad = \quad F^{\sharp^{\epsilon^\sharp}}$$

with $\epsilon^\sharp \leqslant \epsilon$. Let $F^\sharp \restriction I^\sharp$ be the restriction of $F^\sharp$ to its iterates $I^\sharp \triangleq \{F^{\sharp^\delta} \mid 0 \leqslant \delta \leqslant \epsilon^\sharp\}$. Then $F^{\sharp^{\epsilon^\sharp}} = \mathbf{lfp}^{\sqsubseteq} F^\sharp \restriction I^\sharp$ and if $F^\sharp$ is monotone then $F^{\sharp^{\epsilon^\sharp}} = \mathbf{lfp}^{\sqsubseteq} F^\sharp$. □

Proof. We apply Corollary 101 since it is not necessary to assume $F^\sharp$ to be monotone for these iterates to be increasing since they are the image of an increasing chain by the monotone α.

By the commutation property and definition of ϵ, $F^\sharp(F^{\sharp^\epsilon}) = F^\sharp(\alpha(F^\epsilon)) = \alpha \circ F(F^\epsilon) = \alpha(F^\epsilon) = F^{\sharp^\epsilon}$, proving $\epsilon^\sharp \leq \epsilon$.

We have $F^{\sharp^{\epsilon^\sharp}} = \mathbf{lfp}^{\sqsubseteq} F^\sharp \restriction I^\sharp$ since $F^{\sharp^{\epsilon^\sharp}}$ is the only fixpoint of $F^\sharp$ on its iterates. In general, $F^{\sharp^{\epsilon^\sharp}} \neq \mathbf{lfp}^{\sqsubseteq} F^\sharp$ as shown by the following counterexample



However if $F^\sharp$ is monotone and $F^\sharp(x) = x$ then by induction $\forall \delta \leqslant \epsilon^\sharp : F^{\sharp^\delta} \sqsubseteq x$ so $F^{\sharp^{\epsilon^\sharp}} = \mathbf{lfp}^{\sqsubseteq} F^\sharp$. □

Example 105 (*Fixpoint Abstraction at a Point*). Continuing Example 97, let $\langle L, \sqsubseteq, \bot, \top\rangle$ be a poset with infimum $\bot$ and supremum $\top$, $x \in L$, S be a set and $F = \boldsymbol{\lambda} \phi \bullet \boldsymbol{\lambda} z \bullet f(z, \phi(z))$ where $f \in (S \times L) \mapsto L)$ is such that $F \in (S \mapsto L) \mapsto (S \mapsto L)$ is monotone and the iterates of F are well defined. Then $\alpha^x(\mathbf{lfp}^{\sqsubseteq} F) = \mathbf{lfp}^{\sqsubseteq} \boldsymbol{\lambda} X \bullet f(x, X)$.

Proof. We apply Corollary 104 to F and discover $F^\sharp = \boldsymbol{\lambda} X \bullet f(x, X)$ by calculus $\alpha^x(F(\phi))$

$$= \quad \alpha^x(\boldsymbol{\lambda} z \bullet f(z, \phi(z))) \quad = \quad f(x, \phi(x)) \qquad \text{\{def. } F \text{ and } \alpha^x\text{\}}$$
$$= \quad f(x, \alpha^x(\phi)) \qquad \text{\{def. } \alpha^x \text{ so we let } F^\sharp = \boldsymbol{\lambda} X \bullet f(x, X)\text{\}}. \quad \square \quad \square$$

The particular case [40, Th. 2] is

Corollary 106. *If $\langle P, \preceq, 0, \vee\rangle$ is a cpo, $\langle P, \preceq\rangle \xleftarrow[\alpha]{\gamma} \langle Q, \sqsubseteq\rangle$, $F \in P \mapsto P$ is monotone, $F^\sharp \in Q \mapsto Q$ and the* commutation property

$$\forall x \in P : x \preceq \mathbf{lfp}^{\preceq} F \Longrightarrow \alpha \circ F(x) = F^\sharp \circ \alpha(x)$$

holds, then $\forall \delta : \alpha(F^\delta) = F^{\sharp^\delta}$, $\mathbf{lfp}^{\preceq} F$ as well as $\mathbf{lfp}^{\sqsubseteq} F^\sharp$ do exist such that

$$\alpha(\mathbf{lfp}^{\preceq} F) \quad = \quad \mathbf{lfp}^{\sqsubseteq} F^\sharp$$

and the iteration order $\epsilon^\sharp$ of $F^\sharp$ is less than or equal to that ϵ of F. If $\langle P, \preceq\rangle \xleftarrow[\alpha]{\gamma} \langle Q, \sqsubseteq\rangle$ then we can choose $F^\sharp = \alpha \circ F \circ \gamma$.

Proof. We apply Corollary 104. The iterates for a monotone F do exist in a cpo. If $\langle P, \preceq\rangle \xleftarrow[\alpha]{\gamma} \langle Q, \sqsubseteq\rangle$ then $\gamma \circ \alpha = \mathbb{1}_Q$ so $\alpha \circ F(x) = \alpha \circ F \circ \gamma \circ \alpha(x) = F^\sharp \circ \alpha(x)$.

Example 107 (*Reachable States*). Let $\langle \Sigma,\ \tau\rangle$ be a transition system (where Σ is a nonempty set of states and $\tau \in \wp(\Sigma \times \Sigma)$ is a transition relation). The *reachable states* from initial states $I \subseteq \Sigma$ by τ is the right/post-image of I by $\tau^\star$ that is $\text{post}[\tau^\star]I$ where $\text{post} \in \wp(\Sigma) \mapsto \wp(\Sigma)$ is $\text{post}[r]X \triangleq \{s' \in \Sigma \mid \exists s \in X : \langle s,\ s'\rangle \in r\}$. We have

$$\text{post}[\tau^\star]I \quad = \quad \mathbf{lfp}^{\subseteq}\,\overline{F} \qquad \text{where} \qquad \overline{F} \quad \triangleq \quad \boldsymbol{\lambda} X \bullet I \cup \text{post}[\tau]X \tag{A.1}$$

where the iterates of $\overline{F}$ satisfy $\forall \delta \leq \omega : \overline{F}^{\delta} = \text{post}[r^{\delta\star}]I$.

Proof. We apply Corollary 106 to $\tau^\star = \mathbf{lfp}^{\subseteq} F$ with $F = \boldsymbol{\lambda} x \bullet \mathbb{1}_{\Sigma} \cup (x \circ \tau)$ with abstraction $\alpha \triangleq \boldsymbol{\lambda} r \bullet \text{post}[r]I$ such that $\langle \wp(\Sigma \times \Sigma),\ \subseteq\rangle \xleftrightarrow[\alpha]{\gamma} \langle \wp(\Sigma),\ \subseteq\rangle$ using the commutation condition $\alpha \circ F = \overline{F} \circ \alpha$ to design the abstract transformer $\overline{F}$. We have $\alpha \circ (\boldsymbol{\lambda} x \bullet \tau^0 \cup x \circ \tau) = \boldsymbol{\lambda} x \bullet \text{post}[\tau^0]I \cup \text{post}[x \circ \tau]I$ by def. $\circ$, α preserves lubs, and def. α.

— $\text{post}[\tau^0]I = \{s' \mid \exists s \in I : \langle s, s'\rangle \in \{\langle s, s\rangle \mid s \in S\}\} = I$ $\qquad$ ≀def. post, $\tau^0 = \mathbb{1}_{\Sigma}, \in$≀

— $\text{post}[x \circ \tau]I \ = \ \{s' \mid \exists s \in I : \exists s'' \in S : \langle s, s''\rangle \in x \wedge \langle s', s''\rangle \in \tau\}$ $\qquad$ ≀def. post, $\circ$, & $\in$≀

$= \{s' \mid \exists s'' \in S : s'' \in \{s'' \mid \exists s \in I : \langle s, s"\rangle \in x\} \wedge \langle s', s''\rangle \in \tau\}$ $\qquad$ ≀commutativity of $\exists$ and def. $\in$≀

$= \text{post}[\tau](\alpha(x))$ $\qquad$ ≀def. post and α≀. □ □

The Galois connection hypothesis can be weaken into a continuity hypothesis on the abstraction α. For example

Corollary 108. *If $\langle P,\ \preceq,\ 0,\ \vee\rangle$ is a poset with infimum 0, $F \in P \mapsto P$ is monotone, the iterates of F are well defined with iteration order ϵ less than or equal to ω,*[8] *$\langle Q,\ \sqsubseteq,\ \perp,\ \sqcup\rangle$ is a poset with infimum $\perp$, $F^\sharp \in Q \mapsto Q$, the abstraction function $\alpha \in P \mapsto Q$ is strict ($\alpha(0) = \perp$), continuous and satisfies the* commutation property

$$\forall \delta \in \mathbb{O} : \alpha \circ F(F^{\delta}) = F^\sharp \circ \alpha(F^{\delta}),$$

then $\forall \delta \leqslant \omega : \alpha(F^{\delta}) = F^{\sharp\delta}$ and

$$\alpha(\boldsymbol{lfp}^{\preceq} F) \quad = \quad F^{\sharp\epsilon^\sharp}$$

with $\epsilon^\sharp \leqslant \epsilon \leqslant \omega$. Let $F^\sharp \restriction I^\sharp$ be the restriction of $F^\sharp$ to its iterates $I^\sharp \triangleq \{F^{\sharp\delta} \mid 0 \leqslant \delta \leqslant \epsilon^\sharp\}$. Then $F^{\sharp\epsilon^\sharp} = \boldsymbol{lfp}^{\sqsubseteq} F^\sharp \restriction I^\sharp$ and if $F^\sharp$ is monotone then $F^{\sharp\epsilon^\sharp} = \boldsymbol{lfp}^{\sqsubseteq} F^\sharp$.

Proof. By definition of the iterates and induction, we have $\forall \delta \in \mathbb{O} : \alpha(F^{\delta}) = F^{\sharp\delta}$ by strictness for the basis $\delta = 0$, by induction hypothesis and commutation property for $0 < \delta < \omega$, by induction hypothesis and for $\delta = \omega$ and $\forall \delta \geqslant \omega, F^{\delta} = F^{\omega}$ since $\epsilon \leq \omega$.

The proof then follows that of Corollary 104.

Theorem 109. *If $\langle P,\ \preceq,\ \vee\rangle$ is a poset, $F \in P \mapsto P$ is continuous, $\langle Q,\ \sqsubseteq,\ \sqcup\rangle$ is a poset, $F^\sharp \in Q \mapsto Q$, $\langle P,\ \preceq\rangle \xleftrightarrow[\alpha]{\gamma} \langle Q,\ \sqsubseteq\rangle$ and $\alpha \circ F = F^\sharp \circ \alpha$ then $F^\sharp$ is continuous.*

Proof. Let x_i, $i \in \mathbb{N}$ be a $\sqsubseteq$-increasing chain of elements of Q. We have

$$\bigsqcup_{i\in\mathbb{N}} F^\sharp(x_i) \ = \ \alpha\left(\bigvee_{i\in\mathbb{N}} F(\gamma(x_i))\right) \qquad \wr\alpha \circ \gamma = \mathbb{1}_Q\text{, def }\circ\text{, commutation, }\alpha\text{ preserves lubs}\wr$$

$$= \alpha\left(F\left(\bigvee_{i\in\mathbb{N}} \gamma(x_i)\right)\right) \qquad \wr\gamma\text{ monotone, so }\gamma(x_i), i \in \mathbb{N}\text{ is an increasing chain, and }F\text{ continuous}\wr$$

$$= F^\sharp\left(\bigsqcup_{i\in\mathbb{N}} x_i\right) \qquad \wr\text{commutation, }\alpha\text{ preserves lubs, }\alpha \circ \gamma = \mathbb{1}_Q\text{, def }\circ\wr. \quad \square$$

[8] ω is the first infinite limit ordinal. An example is when $F \in P \mapsto P$ is continuous.

References

[1] D. Knuth, A generalization of Dijkstra's algorithm, Information Processing Letters 6 (1) (1977) 1–5.
[2] G. Ramalingam, T. Reps, An incremental algorithm for a generalization of the shortest-path problem, Journal of Algorithms 21 (2) (1996) 267–305.
[3] J. Bar-Hillel, M. Perles, E. Shamir, On formal properties of simple phrase structure grammars, Zeitschrift für Phonetik, Sprachwissenschaft und Kommunikationforschung 14 (1961) 143–172.
[4] A. Aho, J. Ullman, Parsing, The Theory of Parsing, Translation and Compiling, vol. 1, Prentice-Hall, Inc., Englewood Cliffs, 1972.
[5] U. Möncke, R. Wilhelm, Iterative algorithms on grammar graphs, in: H. Schneider, H. Gottler (Eds.), Proceedings of the Eight Conference on Graphtheoretic Concepts in Computer Science, WG'82, Hanser Verlag, Munchen, 1982, pp. 177–194.
[6] U. Möncke, Generierung von systemen zur transformation attributierter operatorbäume; komponenten des systems und mechanismen der generierung, Diplomarbeit, Universität des Saarlandes, Saarbrücken, 1985.
[7] U. Möncke, R. Wilhelm, Grammar flow analysis, in: H. Alblas, B. Melichar (Eds.), Proceedings of the Attribute Grammars, Applications and Systems, International Summer School SAGA, Prague, 4–13 June, 1991, in: Lecture Notes in Computer Science, vol. 545, Springer, Heidelberg, 1991, pp. 151–186.
[8] R. Wilhelm, D. Maurer, Übersetzerbau. Theorie, Konstruktion, Generierung, Springer, Heidelberg, 1992.
[9] P. Cousot, R. Cousot, Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints, in: Conference Record of the Fourth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, ACM Press, New York, Los Angeles, 1977, pp. 238–252.
[10] N. Chomsky, Three models for the description of language, IEEE Transactions on Information Theory 2 (3) (1956) 113–124.
[11] N. Chomsky, de Gruyter, Mouton, 1957.
[12] A. Oettinger, Automatic syntactic analysis and the pushdown store, in: Proceedings of Symposia in Applied Math. 12.
[13] H. Rutishauser, Automatische Rechenplanfertigung bei programmgesteuerten Rechenmaschinen, Zeitschrift für Angewandte Mathematik und 32 (1952) 312–313. mitt. Nr. 3 aus dem Inst. f. Angew. Math. der ETH Zürich, 45 p., 1952, Birkhäuser, Basel.
[14] C. Boehm, Calculatrices digitales. du déchiffrage de formules logico-mathématiques par la machine même dans la conception du programme. dissertation eth zürich (1952), Annali di Matematica Pura ed Applicata 4 (37) (1954) 5–47.
[15] F. Bauer, K. Samuelson, Verfahren zur automatischen Verarbeitung von kodierten Daten und Rechenmaschine zur Ausübung des Verfahrens, deutsches Patentamt, Auslegeschrift 1094019, B441221X/42m. Anmeldetag: 30. März 1957. Bekanntmachung der Anmeldung und Ausgabe der Auslegeschrift: 1.Dezember 1960. Dr. Friedrich Ludwig Bauer und Dr. Klaus Samuelson, München, sind als Erfinder genannt worden. Erteilt 12.8.1971, DE-PS 1094019, 1957.
[16] K. Samuelson, F. Bauer, Sequentielle formelübersetzung, Elektronische Rechenanlagen 1 (4) (1959) 176–182.
[17] F. Bauer, From the stack principle to ALGOL, in: M. Broy, E. Denert (Eds.), Software Pioneers: Contributions to Software Engineering, Springer, Heidelberg, 2002, pp. 26–42.
[18] P. Naur, The European side of the last phase of the development of ALGOL 60, in: R. Wexelblat (Ed.), History of Programming Languages I, ACM Press, New York, 1978, pp. 92–139.
[19] N. Chomsky, Context free grammars and pushdown storage, quarterly Progress Report 65, MIT Research Laboratory in Electronics, Cambridge, 1962.
[20] J. Evey, Application of pushdown store machines, in: Proceedings of the 1963 Fall Joint Computer Conference, AFIPS Press, Montreal, 1963.
[21] M. Schützenberger, On context free languages and pushdown automata, Information and Control 6 (1963) 246–264.
[22] H. Hoogeboom, J. Engelfriet, Pushdown automata, in: C. Martín-Vide, V. Mitrana, G. Paun (Eds.), Formal Languages and Applications, in: Studies in Fuzziness and Soft Computing, vol. 148, Springer, Heidelberg, 2004, pp. 117–138.
[23] J. Autebert, J. Berstel, L. Boasson, Context-free languages and pushdown automata, in: G. Rozenberg, A. Salomaa (Eds.), Word, Language, Grammar, in: Handbook of Formal Languages, vol. 1, Springer, Heidelberg, 1997.
[24] S. Ginsburg, The Mathematical Theory of Context-Free Languages, McGraw-Hill, New York, 1966.
[25] P. Cousot, Constructive design of a hierarchy of semantics of a transition system by abstract interpretation, Theoretical Computer Science 277 (1—2) (2002) 47–103.
[26] P. Cousot, Semantic foundations of program analysis, invited chapter, in: S. Muchnick, N. Jones (Eds.), Program Flow Analysis: Theory and Applications, Prentice-Hall, Inc., Englewood Cliffs, 1981, pp. 303–342 (Chapter 10).
[27] P. Cousot, R. Cousot, Systematic design of program analysis frameworks, in: Conference Record of the Sixth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, ACM Press, New York, San Antonio, 1979, pp. 269–282.
[28] P. Cousot, R. Cousot, Constructive versions of Tarski's fixed point theorems, Pacific Journal of Mathematics 82 (1) (1979) 43–57.
[29] S. Ginsburg, G. Rice, Two families of languages related to ALGOL, Journal of the Association for Computing Machinery 9 (1962) 350–371.
[30] M. Schützenberger, On a theorem of R. Jungen, Proceedings of the American Mathematical Society 13 (1962) 885–889.
[31] N. Chomsky, M. Schützenberger, The algebraic theory of context-free languages, in: P. Bradford, D. Hirschberg (Eds.), Computer programming and Formal Systems, North-Holland Pub. Co., Amsterdam, 1963, pp. 118–161.
[32] A. Aho, R. Sethi, J. Ullman, Compilers. Principles, Technique and Tools, Addison-Wesley Pub. Co., Reading, 1986.
[33] E. Dijkstra, A note on two problems in connexion with graphs, Numerische Mathematik 1 (1959) 269–271.
[34] D. Hays, Introduction to Computational Linguistics, American Elsevier, New York, 1967.
[35] D. Younger, Recognition and parsing of context-free languages in time n^3, Information and Control 10 (2) (1967) 609–617.
[36] T. Kasami, An efficient recognition and syntax analysis algorithm for context-free languages, Tech. Rep., Air Force Cambridge Research Laboratory, Bedford, August 1965.
[37] J. McCarthy, J. Painter, Correctness of a compiler for arithmetic expressions, in: J. Schwartz (Ed.), Proceedings of the Symposium in Applied Mathematics, in: Mathematical Aspects of Computer Science, vol. 19, American Mathematical Society, Providence, 1967, pp. 33–41.
[38] X. Leroy, Formal certification of a compiler back-end or: programming a compiler with a proof assistant, in: Conference Record of the Thirtythird Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, ACM Press, New York, Charleston, 2006, pp. 42–54.
[39] X. Leroy, Coinductive big-step operational semantics, in: P. Sestoft (Ed.), Proceedings of the Fifteenth European Symposium on Programming Languages and Systems, in: Lecture Notes in Computer Science, vol. 3924, Springer, Heidelberg, 2006, pp. 54–68.
[40] P. Cousot, The calculational design of a generic abstract interpreter, invited chapter, in: M. Broy, R. Steinbrüggen (Eds.), Calculational System Design, in: NATO Science Series, Series F: Computer and Systems Sciences, vol. 173, IOS Press, Amsterdam, 1999, pp. 421–505.
[41] D. Pichardie, Interprétation abstraite en logique intuitionniste: extraction d'analyseurs Java certifiés, Ph.D. Thesis, Université de Rennes I, December 2005.
[42] P. Cousot, R. Cousot, R. Giacobazzi, Abstract interpretation of resolution-based semantics, Theoretical Computer Science 410 (46) (2009) 4724–4746.
[43] G. Păun, Marcus Contextual Grammars, Studies in Linguistics and Philosophy, Kluwer Academic Publishers, Dordrecht, 1997.
[44] A. Ehrenfeucht, G. Păun, G. Rozenberg, Contextual grammars and formal languages, in: G. Rozenberg, A. Salomaa (Eds.), Handbook of Formal Languages, vol. 2, Springer, Heidelberg, 1997, pp. 237–293.
[45] P. Boullier, From contextual grammars to range concatenation grammars, Electronic Notes in Theoretical Computer Science 53 (2001) 41–52. http://www.elsevier.nl/locate/entcs/volume53.html.
[46] B. Davey, H. Priestley, Introduction to Lattices and Order, second ed., Cambridge University Press, Cambridge, 2002.